PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G09G 5/36 — A1

(11) International Publication Number: WO 96/36037

(43) International Publication Date: 14 November 1996 (14.11.96)

(21) International Application Number: PCT/US96/06438

(22) International Filing Date: 8 May 1996 (08.05.96)

(30) Priority Data:
08/438,630 10 May 1995 (10.05.95) US

(71) Applicant: THE 3DO COMPANY [US/US]; 600 Galveston Drive, Redwood City, CA 94063 (US).

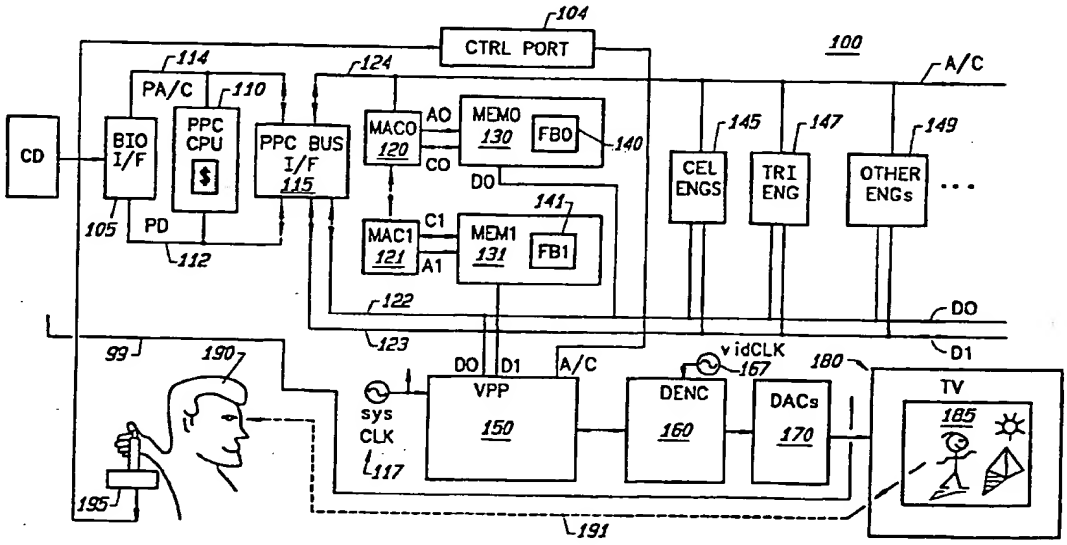
(72) Inventors: THAIK, Richard, W.; 1566 Cleo Springs Drive, San Jose, CA 95131 (US). MICAL, Robert, Joseph; 25 Geri Place, Redwood City, CA 94063 (US). LANDRUM, Stephen, Harland; 179 Haas Avenue #7, San Leandro, CA 94577 (US). WASSERMAN, Steve, C.; 10455 North Blaney Avenue, Cupertino, CA 95014 (US). NEEDLE, David, Lewis; 2981 Northwood Drive, Alameda, CA 94501 (US). CARROLL, Tina, L.; 1055 Manet Drive #38, Sunnyvale, CA 94087 (US).

(74) Agent: WOLFELD, Warren, S.; Fliesler, Dubb, Meyer and Lovejoy, Suite 400, Four Embarcadero Center, San Francisco, CA 94111-4156 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: VIDEO DISPLAY SYSTEM HAVING BY-THE-LINE AND BY-THE-PIXEL MODIFICATION



(57) Abstract

Method and apparatus (150) for managing color modification of a raster based image on a real time, line-by-line basis, or pixel-by-pixel basis in response to pixel information (140/141) and control information (142/143) stored in memory (130/131).

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia
AT	Austria
AU	Australia
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CS	Czechoslovakia
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgystan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakhstan
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam

VIDEO DISPLAY SYSTEM HAVING BY-THE-LINE AND BY-THE-PIXEL MODIFICAITON

BACKGROUND

1. Field of the Invention

The invention relates generally to digital image processing and the display of digitally generated images.

The invention relates more specifically to the problem of creating raster-based, high-resolution animated images in real time, where the mechanism for generating each raster line is modifiable on a by-the-pixel basis, or on a by-the-line basis, or on a by-a-group of lines basis.

2. Cross Reference to Related Applications

This application is related to the following co-pending applications:

(a) PCT Patent Application Serial No. PCT/US92/09342, entitled RESOLUTION ENHANCEMENT FOR VIDEO DISPLAY USING MULTI-LINE INTERPOLATION, by inventors Mical et al., filed November 2, 1992, [Attorney Docket No. MDIO3050,] and also to U.S. Patent Application Serial No. 07/970,287, bearing the same title, same inventors and also filed November 2, 1992;

(b) PCT Patent Application Serial No. PCT/US92/09349, entitled AUDIO/VIDEO COMPUTER ARCHITECTURE, by inventors Mical et al., filed November 2, 1992, [Attorney Docket No. MDIO4222,] and also to U.S. Patent Application Serial No. 07/970,308, bearing the same title, same inventors and also filed November 2, 1992;

(c) PCT Patent Application Serial No. PCT/US92/09350, entitled METHOD FOR CONTROLLING A

SPRYTE RENDERING PROCESSOR, by inventors Mical et al., filed November 2, 1992, [Attorney Docket No. MDIO3040,] and also to U.S. Patent Application Serial No. 07/970,278, bearing the same title, same inventors and also filed November 2, 1992;

(d) PCT Patent Application Serial No. PCT/US92/09462, entitled SPRYTE RENDERING SYSTEM WITH IMPROVED CORNER CALCULATING ENGINE AND IMPROVED POLYGON-PAINT ENGINE, by inventors Needle et al., filed November 2, 1992, [Attorney Docket No. MDIO4232,] and also to U.S. Patent Application Serial No. 07/970,289, bearing the same title, same inventors and also filed November 2, 1992;

(e) PCT Patent Application Serial No. PCT/US92/09460, entitled METHOD AND APPARATUS FOR UPDATING A CLUT DURING HORIZONTAL BLANKING, by inventors Mical et al., filed November 2, 1992, [Attorney Docket No. MDIO4250,] and also to U.S. Patent Application Serial No. 07/969,994, bearing the same title, same inventors and also filed November 2, 1992;

(f) PCT Patent Application Serial No. PCT/US92/09467, entitled IMPROVED METHOD AND APPARATUS FOR PROCESSING IMAGE DATA, by inventors Mical et al., filed November 2, 1992, [Attorney Docket No. MDIO4230,] and also to U.S. Patent Application Serial No. 07/970,083, bearing the same title, same inventors and also filed November 2, 1992;

(g) PCT Patent Application Serial No. PCT/US94/12521, entitled DISPLAY LIST MANAGEMENT MECHANISM FOR REAL-TIME CONTROL OF BY-THE-LINE MODIFIABLE VIDEO DISPLAY SYSTEM, by inventors Robert Joseph Mical et al., filed November 1, 1994, [Attorney Docket No. MDIO4255,] and also to U.S. Patent Application Serial No. 08/146,505, bearing the same title, same inventors and filed November 1, 1993; and

(h) U.S. Patent Application Serial No. 08/311,192 entitled REAL TIME DECOMPRESSION AND POST-DECOMPRESS MANIPULATION OF COMPRESSED FULL MOTION VIDEO, by inventors Steve C. Wasserman et al., filed september 23, 1994 [Attorney Docket No. MDIO4370].

The related patent applications are all commonly assigned with the present application and are all incorporated herein by reference in their entirety.

The present application is to be considered a continuation-in-part of at least the above cited U.S. Patent Application Serial No. 08/146,505, entitled DISPLAY LIST MANAGEMENT MECHANISM FOR REAL-TIME CONTROL OF BY-THE-LINE MODIFIABLE VIDEO DISPLAY SYSTEM, by inventors Robert Joseph Mical et al., filed November 1, 1993, [Attorney Docket No. MDIO4255].

3. Description of the Related Art

In recent years, the presentation and pre-presentation processing of visual imagery has shifted from what was primarily an analog electronic format to an essentially digital format.

Unique problems come to play in the digital processing of image data and the display of such image data.

The more prominent problems include providing adequate storage capacity for digital image data and maintaining acceptable data throughput rates while using hardware of relatively low cost. In addition, there is the problem of creating a sense of realism in digitally generated imagery, particularly in animated forms of such imagery.

Visual realism for imagery generated by digital video game systems, by simulators and the like can be enhanced by providing special effects such as, but not limited to, creating moving sprites, making real-time

changes in the shadowing and/or highlighting of various objects, smoothing or sharpening the contours of various objects at different times, and so forth.

Visual realism can be further enhanced by projecting 3-dimensional object definitions onto a 2-dimensional display screen and rotating, scaling or otherwise manipulating the 3-dimensional object definitions in real time prior to their projection onto the 2-dimensional display screen.

Visual realism can be additionally enhanced by increasing the apparent resolution of a displayed image so that it has a smooth photography-like quality rather than a grainy disjoined-blocks appearance of the type found in low-resolution computer-produced graphics of earlier years.

Visual realism can be even further enhanced by increasing the total number of different colors and/or shades in each displayed frame of an image so that, in regions where colors and/or shades are to change in a smooth continuum by subtle degrees of hue/intensity, the observer perceives such a smooth photography-like variation of hue/intensity rather than a stark and grainy jump from one discrete color/shade to another.

Although bit-mapped computer images originate as a matrix of discrete lit or unlit pixels, the human eye can be fooled into perceiving an image having the desired photography-like continuity if the displayed matrix of independently-shaded (and/or independently colored) pixels has dimensions of approximately 500-by-500 pixels or better at the point of display and a large variety of colors and/or shades on the order of roughly 24 bits-per-pixel or better.

The VGA graphics standard, which is used in many present-day low-cost computer systems, approximates this effect with a display matrix having dimensions of

640-by-480 pixels. However, conventional low-cost VGA graphic systems suffer from a limited per-frame palette of available colors and/or shades.

Standard NTSC broadcast television systems also approximate the continuity mimicking effect by using interlaced fields with 525 lines per pair of fields and a horizontal scan bandwidth (analog) that is equivalent to approximately 780 RGB colored dots per line (approximately 480-by-640 active pixels).

More advanced graphic display standards such as Super-VGA and High Definition Television (HDTV) rely on much higher resolutions, 1024-by-786 pixels per frame, for example.

It is expected that display standards will emerge in the future with yet higher resolution numbers (e.g., 2048-by-2048). It is expected that the number of bits per displayed pixel will similarly increase in the future.

With each increase in frame resolution (pixels per frame) and each increase in the number of colors and/or shades that are available per display frame, the problem of providing adequate storage capacity for the corresponding digital image data becomes more acute. The problem of providing sufficient data processing throughput rates also becomes more acute. This is particularly so if an additional constraint is imposed of keeping hardware costs within an acceptable price versus performance range.

A display with 640-by-480 independent pixels (307,200 pixels total) calls for a conventional frame buffer having at least 19 address bits as its input or a corresponding 2^{19} independently-addressable data words (= 512K words). Each data word in such a frame buffer stores a binary code representing the shading and/or color of an individual pixel. Each doubling of

display resolution, say from 640-by-480 pixels to 1280-by-960 pixels, calls for a four-fold increase in the storage capacity of the frame buffer. Each doubling of per-pixel color/shade variation, say from 8 bits-per-pixel to 16 bits-per-pixel, calls for an additional two-fold increase in storage capacity.

This means that a system designed according to conventional techniques to initially render an at-display image of 640-by-480 independent pixels per frame, at 8 bits-per-pixel, would conventionally require an eight-fold increase of memory capacity, from 512K bytes to 4MB (four Megabytes) as a result of doubling each of the number of pixels per row, number of pixels per column and the number of bits-per-pixel.

In cases where parts or all of the resultant 1280-by-960 display field have to be modified in real-time (to create a sense of animation), the eight-fold increase of storage capacity calls for a corresponding eight-fold increase in data processing bandwidth (image bits processed per second) as compared to what was needed for processing the original, 8 bits-per-pixel, 640-by-480 pixels frame.

At some point, the benefit versus cost ratio associated with such an approach has to be questioned. Is it necessary to continuously increase storage capacity and to forever seek faster data processing speeds in order to obtain improvements in apparent performance for the end user?

Perhaps a given increase in performance is not worth the corresponding increase in system cost if the conventional approach is followed of increasing storage capacity and providing a corresponding increase in data processing speed.

Perhaps it is possible to create a perception of improved performance without suffering a concomitant burden of significantly higher system cost.

Such an objective can be realized by using a High-performance, Inexpensive, Image-Rendering system (HI-IR system) such as disclosed in the above cited set of co-related patent applications.

Part of the low-cost and high-performance (high performance/price ratio) of the earlier-disclosed HI-IR system is owed to the use, in a display-defining path of the system, of Color LookUp Tables (CLUT's) whose color-mapping functions are modifiable on a by-the-line basis and even on a by-the-pixel basis.

Another part of the high performance/price ratio of the HI-IR system is owed to the use, in the display-defining path of the system, of a subposition-weighted Interpolator whose subposition weights are modifiable on a by-the-pixel basis and whose modes of operation (horizontal-interpolation on/off and vertical-interpolation on/off) are modifiable on a by-the-line or by-the-frame basis.

Yet another part of the low-cost and high-performance of the HI-IR system is owed to the use, in a bitmap-defining portion of the system, of a unique set of one or more 'spryte' rendering engines (also called cel-animating engines) for executing a list of bitmap modification instructions stored in a queue.

A description of such a spryte-generating mechanism may be found in the above cited PCT Patent Application Serial No. PCT/US92/09350, entitled METHOD FOR CONTROLLING A SPRYTE RENDERING PROCESSOR, and also in PCT Patent Application Serial No. PCT/US92/09462, entitled SPRYTE RENDERING SYSTEM WITH IMPROVED CORNER CALCULATING ENGINE AND IMPROVED POLYGON-PAINT ENGINE.

Yet a further part of the high performance/price ratio of the HI-IR system is owed to the use, in an application program interface portion of the system, of a flexible yet robust list-based control of the post-frame buffer 'display path' segments of the system.

A description of such a list-based mechanism may be found in the above cited PCT Patent Application Serial No. PCT/US94/12521, entitled DISPLAY LIST MANAGEMENT MECHANISM FOR REAL-TIME CONTROL OF BY-THE-LINE MODIFIABLE VIDEO DISPLAY SYSTEM.

More recently, the industry has sought ways to add even more realism to the rendered images that originate from computer-generated image definitions.

Compound systems are being proposed that have both real-time 2-dimensional object manipulation means (sprite manipulation means) and real-time 3-dimensional object manipulation means as well as other means that contend for access to system memory (e.g., system CPU's and real-time sound generating modules).

The compound nature of such systems places a strain on system memory to deliver (or store) time-critical data to (or from) devices or modules that need to operate on a real-time basis. An example of time-critical data is video data that may be needed on a real-time basis, within the download time window of a horizontal raster line for example, in order to provide real-time rendering of an interactive game image or an interactive simulator image.

The proposed compound systems have added functionalities that further strain the throughput capabilities (data bandwidth) of the memory-access management subsystem and complicate the tasks of the memory-access management subsystem. The memory-access management subsystem now needs to arbitrate among a larger number of contenders for memory access.

The added functionalities of the proposed compound systems additionally strain the throughput capabilities and complicate the tasks of any system CPU's that have to supervise the activities of the image manipulation means on a real-time basis. (The term "CPU's" refers here to a general-purpose data processing subsystem which may be implemented either in a centralized unit format, as for example a single truly-central processing unit; or which may be implemented in a plural units format, such as in a parallel processing system.)

A system architecture is needed for reducing contention among plural potential requesters for system memory access.

A system architecture is needed for reducing contention by plural software modules for access to the limited resources of system CPU's.

A methodology is needed for simultaneously satisfying the needs of multiple, time-critical processes such as those of a real-time video display subsystem and those of a real-time animation subsystem.

SUMMARY OF THE INVENTION

The above-mentioned problems are overcome in accordance with the invention by providing a graphics system that has a multi-port memory subsystem storing video display control lists (VDL's) and a real-time programmably-configurable video post-processor (VPP) coupled to the memory subsystem and responsive to the stored VDL's.

A video post-processor (VPP) in accordance with the invention reduces the load on system memory and on the system CPU's. Such a VPP periodically fetches render-control lists (VDL's) and image data from the memory subsystem on a block basis and processes the

fetched information without requiring continuous real-time intervention by system CPU's. This frees the system CPU (or CPU's) for managing other time-critical tasks of the graphics system.

One embodiment of a video post-processor (VPP) in accordance with the invention includes an addressable register set that is programmable by one or both of a system CPU and a memory-resident control-list.

One embodiment of a graphics system in accordance with the invention comprises an Image-Enhancing And Rendering Subsystem (I-EARS) that includes the video post-processor (VPP). The programmable register set of the VPP defines primitive configurations for the VPP and for remaining portions of the I-EARS. The register-controlled primitives of the I-EARS include those that are modifiable on a per-pixel (PP) basis, those that are modifiable on a per-scanline (PS) basis, and those that are modifiable on a per-field (PF) basis.

The one embodiment of the video post-processor (VPP) further includes a FIFO means for fetching sufficient pixel-defining data on a timely basis from a memory subsystem so as to allow the configurable image-enhancing and rendering subsystem (I-EARS) to continuously supply video signals on a timely basis to a real-time video display means.

Each pixel-defining entity that is stored within the VPP FIFO means can include a so-called D-bit that may be used for modifying operations of the I-EARS on a per-pixel basis.

A variety of windowing effects can be provided for by modulating the D-bit and enabling it to define on a by-the-pixel basis whether various options within the image-enhancing and rendering subsystem (I-EARS) are to be enabled, including but not limited to:

(a) horizontal interpolation, (b) vertical interpolation, (c) horizontal filtering, (d) ordered dithering, and (e) RGB-to-YUV conversion.

These and other features of the graphics system will become clearer by referring to the below detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

The below detailed description makes reference to the accompanying drawings, in which:

FIGURE 1 is a block diagram of a High-performance, Inexpensive, Image-Rendering system (HI-IR system) in accordance with the invention that includes a programmably-configurable Video Post Processor (VPP);

FIGURE 2 is a block diagram showing more details of a video post-processor (VPP) in accordance with the invention;

FIGURE 3A shows three different formats according to which frame buffer data may be stored in system memory;

FIGURE 3B shows the FIFO data format for split mode and the FIFO input and output sections;

FIGURE 4A illustrates post-FIFO manipulation of video data by plural color look-up tables (CLUT's), by a CLUT-bypass circuit, and by a blue-pen bit substitution circuit responsive to the D-bit; and

FIGURE 4B illustrates post-CLUT processing of video data by an interpolator circuit.

DETAILED DESCRIPTION

Referring to Fig. 1, a block diagram of an image processing and display system 100 in accordance with the invention is shown.

A key feature of system 100 is that it is relatively low in cost and yet it provides mechanisms

for handling complex image scenes in real time and displaying them such that they appear to have relatively high resolution and a wide variety of colors and/or shades per displayed frame.

This feature is made possible by including within the system 100, an Image-Enhancing And Rendering Subsystem (I-EARS) comprised of: a video post-processor (VPP) 150, a digital encoder (DENC) 160, and a plurality of digital-to-analog converters (DAC's) 170.

The image-enhancing and rendering subsystem (I-EARS) 150-160-170 may be formed within one integrated circuit (IC) chip or it may be distributed across a few IC chips. In one particular embodiment, elements 150, 160 and 170 are implemented within one integrated circuit (IC) chip together with elements 115, 120, 121, 122, 123, 124, 145 and 147 of Fig. 1. This one IC chip is referred to herein as the 'I-EARS chip'.

Within the I-EARS group of elements, 150-160-170, there is provided a set of user-programmable Color LookUp Table modules (CLUT's) 231, 232, and 235 as seen in the more-detailed view of Fig. 2. Also provided, are a hardwired pseudo-linear CLUT circuit 233 which can be programmably substituted for one of CLUT's 231-232 on a by-the-pixel basis, and a user-programmable resolution-enhancing interpolator 250 as further seen in Fig. 2.

(Note that elements referenced by numerals in the '100' number series generally appear in Fig. 1, those referenced by numerals in the '200' number series generally appear in Fig. 2, and so forth.)

The operations and importance of the components within the image-enhancing and rendering subsystem (I-EARS) 150-160-170 may be better appreciated by first

considering the video processing operations of system 100 in an overall sense.

Except as otherwise stated, all or most parts of system 100 are implemented on a single printed circuit board 99 and the circuit components are defined within one or a plurality of integrated circuit (IC) chips mounted to the board 99. Except as otherwise stated, all or most of the circuitry is preferably implemented in CMOS (complementary metal-oxide-semiconductor) technology using 0.5 micron or narrower line widths. An off-board power supply (not shown) delivers electrical power to the board 99.

System 100 includes a real-time video display unit (VDU) 180 such as an NTSC standard television monitor or a PAL standard television monitor or a 640-by-480 VGA computer monitor or a higher-resolution monitor. The VDU 180 is used for displaying high-resolution animated images 185 to a system user 190.

The video display unit (VDU) 180 may also include audio output means (not shown) for simultaneously producing and supplying corresponding multi-phonic or monophonic sound to the system user 190. Alternatively, such audio output means (not shown) is provided separately from the VDU. Although not shown, it is understood that system 100 may include appropriate audio signal generating circuitry for driving the audio output means (not shown) and that the audio signal generating circuitry (not shown) may include digital audio synthesis circuitry for creating effects that give the impression of 3-dimensional sound (echoes, Doppler effect and so forth).

The image-enhancing and rendering subsystem (I-EARS) 150-160-170 drives the video display unit (VDU) 180. A system user 190 observes the displayed imagery 185 and hears any accompanying audio by way of

audio-visual coupling path 191. An interactive response mechanism 195 is provided in the form for example of a joystick and/or a pad of push-buttons for allowing the system user 190 to feedback to the hardware (e.g., CPU 110) his or her real-time responses to the perceived audiovisual show by way of control port interface 104.

The image-enhancing and rendering subsystem (I-EARS) 150-160-170 has a pipelined structure that moves data signals synchronously from an upstream portion to a downstream portion. The downstream portion of the I-EARS 150-160-170 is clocked by a video clock generator (vidCLK) 167 that operates according to pre-specified pixel and control rates of the VDU 180. In one embodiment, the vidCLK rate is approximately 12.5 to 15 MHz.

The digital encoder (DENC) 160 includes a timing section that generates frame synchronization signals such as vertical synch pulses (V-synch) and horizontal synch pulses (H-synch) in synchronism with the vidCLK 167.

The upstream portion of the I-EARS 150-160-170 is clocked by a system clock generator (sysCLK) 117 that operates according to a pre-specified system rate. The system rate is typically different from the vidCLK rate. In one embodiment, the sysCLK rate is approximately 66 MHz. The sysCLK 117 is used for driving other portions of system 100 such as the illustrated CPU 110, cel-engines 145, and triangle engine 147.

System 100 further includes a real-time audiovisual-data processing subsystem comprised of: a control port interface 104, a basic I/O interface module 105, a general purpose central-processing unit (CPU) 110, a multi-port memory 130-131, a plurality of

2-dimensional image manipulating engines (cel-engines) 145, and a 3-dimensional image manipulating engine (triangle engine) 147. The real-time audiovisual-data processing subsystem may include other image-data manipulating engines and audio-data manipulating engines as indicated at 149.

In one embodiment, the memory 130-131 is formed as two independently addressable units, 130 and 131. Units 130 and 131 preferably utilize SDRAM technology (synchronous dynamic random access memory). They may also utilize other high-speed random access data storage technologies such as video-speed static random-access memory subunit (VSRAM).

Access to the first memory unit (MEM0) 130 is managed by a corresponding first memory access control unit (MAC0) 120.

Access to the second memory unit (MEM1) 131 is managed by a respective and independent second memory access control unit (MAC1) 121.

The MAC0 unit 120 may exercise exclusive control over the address (A0) and control (C0) buses of the first memory unit 130. Any other device which wishes to gain access to the A0 and C0 buses then has to send a corresponding request to the MAC0 unit 120 over address/control bus 124. The MAC0 unit 120 arbitrates among contending requestors and grants access to the highest priority requestor.

Similarly, the MAC1 unit 121 may exercise exclusive control over the address (A1) and control (C1) buses of the second memory unit 131. Any other device which wishes to gain access to the A1 and C1 buses then has to send a corresponding request to the MAC1 unit 121 over address/control bus 124. The MAC1 unit 121 arbitrates among contending requestors and grants access to the highest priority requestor.

The first memory unit 130 has a 32-bit wide bidirectional data bus 122 which is also designated as D0. The second memory unit 131 has a 32-bit wide bidirectional data bus 123 which is also designated as D1. Devices such as CPU 110 read and write data into memory units 130 and 131 by way of respective data buses D0 and D1.

If desired, the illustrated, parallel memory architecture may be expanded to include additional memory units (MEM's) such as 130-131 and respective additional memory access control units (MAC's) such as 120-121. The advantage of this parallel memory architecture is, of course, that multiple requestors can obtain simultaneous access to different parts of a centralized memory. It is the responsibility of the operating system software (OS) to see to it that contentions for a same memory unit (e.g., 130 or 131) are minimized.

The operating system (OS) software can be stored in a nonvolatile storage unit such as ROM (not shown) or CD-ROM (not shown) that is operatively coupled to, or forms part of, the system memory 130-131. For high speed nonvolatile storage such as ROM, the OS instructions can be executed directly from that storage. For slower-speed nonvolatile storage such as CD-ROM (compact disk), the OS instructions can be downloaded into system RAM 130-131 and executed from there.

The illustrated embodiment of the image processing and display system 100 uses a PowerPC™ 602 CPU 110 such as made by International Business Machines (IBM) of New York. Other CPU's may of course be used.

The PowerPC™ CPU 110 has its own processor data bus (PD) 112 and its own address/control bus (PA/C)

114. The PowerPC™ data and address/control buses, 112 and 114, couple to the memory data and memory address/-control buses, 122, 123 and 124, by way of a bus interface unit 115. Signals from peripheral devices such as the CD-drives, magnetic disk drives, external memory and the like couple to the PowerPC™ data and address/control buses, 112 and 114, by way of the basic-I/O interface unit 105. An internal cache memory ($) is provided within the PowerPC™ chip for enabling data processing by the CPU 110 at the same time that other devices access the system memory 130-131. Cache coherency is maintained through a snooping mechanism.

Instructions and/or image data are loadable into the memory units 130-131 from a variety of sources (not shown), including but not limited to magnetic or optical floppy or hard disk drives, a CD-ROM drive, a silicon ROM (read-only-memory) device, a cable headend, a wireless broadcast receiver, a telephone modem, etc.

The downloaded instructions are not necessarily for execution by the CPU 110. They may instead be directed to the post-frame buffer, image-enhancing and rendering subsystem (I-EARS) 150-160-170 as will be seen below, or to other instruction executing components such as, but not limited to, the audiovisual data manipulating engines 145-149.

Buses 122, 123 and 124 depict in a general sense the respective data and control paths for moving instructions and image data into and out of memory units 130-131. Downloaded image data can be in compressed or decompressed format. Compressed image data may be temporarily stored in a compressed image buffer of memory 130-131 and expanded into decompressed format on an as needed basis. Displayable image data, such as that provided in a below-described frame buffer

(FB0 140) may be maintained as a linkable list of non-compressed data.

For purposes of a below-described interpolation process, a first image line in a given frame buffer is referenced as an 'upper' line and a vertically subsequent second image line of the frame buffer is referenced as a 'lower' line. The designations of 'upper' and 'lower' are made by software and are swappable. A same image line can be designated at different times as being 'upper' or 'lower'.

The CPU 110 sets one or more 'forced-address' registers (FV0A and FV1A) within the VPP 150 to point to respective locations in the system-memory 130-131 that store the start of a currently active 'Video Display List' (VDL). The VDL has a linked-list structure wherein the start block can point to a second block or back to itself, the second block can point to a third block or back to itself, and so forth. A video display list (VDL) is deemed exhausted when the video post-processor (VPP) 150 has generated a software-defined number of screen lines.

After a 'valid' VDL start location is written into one of the 'forced-address' registers (FV0A and FV1A), and the CPU brings the DENC 160 out of a reset state (using a DVER control register defined below), the DENC begins to issue V-sync and H-sync pulses, and a DMA controller within the VPP 150 responsively begins to periodically send requests to the corresponding MAC (120 or 121) for successive blocks of the active Video Display List (VDL) that contain render control words until the VDL is exhausted. The VPP 150 stores the fetched control words and processes the fetched image data accordingly so as to render a desired image 185 on the video display unit (VDU) 180.

At the same time that the video post-processor (VPP) 150 is repeatedly fetching items from the active video display list (VDL) and rendering the corresponding imagery onto VDU 180, the CPU 110 and/or the other image manipulating engines 145-149 can begin to access binary-coded data stored within the memory 130-131 and to modify the stored data within memory 130-131 at a sufficiently high-rate of speed to create an illusion for an observer 190 that real-time animation is occurring in a high-resolution image 185 (e.g., 640-by-480 pixels, 24 bits-per-pixel) then being displayed on video display unit 180.

In many instances, the observer 190 will be interacting with the animated image 185 by operating buttons or a joystick or other input means of the interactive response mechanism 195. The system user's real-time responses are fed back to the control port interface unit 104 (or directly to the CPU 110) and the receiving unit reacts accordingly in real-time.

The term 'real-time' as used here means sufficiently fast (e.g., within roughly 15 to 0.3 milliseconds) so that a human user 190 perceives the corresponding changes to the audiovisual show emanating from the audio and visual output units (e.g., VDU 180) to be occurring substantially instantaneously.

The image 185 that is rendered on VDU 180 is defined in part by bitmap data stored in one or more screen-band buffers (e.g., 140 and 141) within memory 130-131. Each screen-band buffer contains one or more scanlines of bit-mapped image data. Screen-bands can be woven together in threaded list style to define a full "screen" as will become apparent below when the structure of VDL's is detailed. Alternatively, a single screen-band (a 'simple' panel) can be defined such that the one band holds the bit-mapped image of an

entire screen (e.g., a full set of 240 low-resolution lines or 480 high-resolution lines).

Major animation changes are preferably performed on a multi-buffered screen basis where the contents of a first screen buffer e.g., FB0 140, are displayed while one or more image modifying engines (e.g., the 2-D image manipulating cel-engines 145 and/or the 3-D image manipulating triangle engine 147) operate on the bit-map of a hidden, second or more screen buffers e.g., FB1 141. Then the screen buffers (FB0 and FB1) are swapped so that the previously hidden second buffer becomes the displayed buffer and the previously displayed first buffer (or yet a third buffer) becomes a buffer whose contents are next modified in the background by the image modifying engines. The swappable screen buffers (e.g., 140 and 141) can be respectively stored in independently accessible units (e.g., 130 and 131) of system memory as shown or jointly in a single one of the memory units.

Each of the swappable screen buffers (e.g., 140 and 141) has a respective video display list (VDL) associated with it. Swapping is accomplished by designating as 'active' the VDL of the desired screen buffer.

Each scanline in a woven-together screen-band buffer (e.g., FB0) may contain image data formatted in accordance with any one of a plurality of pre-specified image formats.

In one such format (known as the 1/555 OPERA LR16 format), 5 bits of each 16-bit wide, memory 'halfword' define a red (R) color component value, 5 other bits define a green (G) color component value, 4 or 5 further bits of the same halfword define a blue (B) color component value, and the remaining one or two bits are optionally used for by-the-pixel modulation of

various parameters including those that control a downstream color look-up table (CLUT) section 230, a downstream interpolator 250 (Fig. 2), a downstream filters section 260, and a downstream dithering section 270.

Referring to Fig. 2, there is shown in block diagram format a portion of the system 100 that is referred to here as the 'video display path' (VDP) 200.

The video display path 200 includes a portion of system memory 130/131 that contains one or more frame buffers 140/141 and an associated set of video display control lists (VDL's) 142/143.

The video display path 200 further includes the video post processor (VPP) 150, the digital encoder 160, the DAC's 170, and the video display unit (VDU 180, not shown in Fig. 2).

Control data from an 'active' control list 142/143 is downloaded by way of data buses 122/123 into a control-storing portion of the VPP 150. The control-storing portion of the VPP 150 includes a set of video display list DMA control registers 210, a set of video display path control registers 240 and a so-called 'next-CLUT' 235.

Image data from one of the frame buffers 140/141 flows downstream from the system memory 130/131 through a memory interface circuit (MEM I/F) 205 of the VPP and then through a programmably splittable FIFO 220. Splittable FIFO 220 can be configured to function at its output as a single first-in first-out buffer or it can be configured to function at its output as two first-in first-out buffers that are referred to as the lower-line FIFO 221 and the upper-line FIFO 222. The outputs of dual FIFO's 221, 222 may appear simultaneously on separate output buses of unit 220 or

they may be output in time multiplexed fashion over a shared bus.

The outputs of FIFO 220 flow through a programmably selectable one or more of a plurality of color lookup tables (CLUT's) 230. The color lookup tables 230 include a lower-line CLUT 231, an upper-line CLUT 232 and a CLUT bypass circuit 233. The CLUT bypass circuit 233 may be programmably substituted for either of the lower-line and upper-line CLUT's 231-232 on a pixel-by-pixel basis in response to each pixel's D-bit.

The contents of CLUT's 231-232 are modifiable on a per-scanline basis (PS). The contents of lower-line CLUT 231 are always copied into upper-line CLUT 232 in each time frame corresponding to the horizontal-blanking period (H-BLANK). During a PS modification, after the LL_CLUT contents have been copied into the UL_CLUT 232, the contents of the next-CLUT 235 are subsequently copied into the lower-line CLUT 231, still during the horizontal-blanking period (H-BLANK). Update of the lower-line CLUT 231 is optional. Its contents could be left as is.

The contents of the next-CLUT 235 are also modifiable on a per-scanline basis (PS) but such modification can be carried out even during the horizontal active-scan period (H-SCAN) since real-time video data does not have to flow through the next-CLUT 235.

(In an alternate embodiment, the designations of 'next', 'lower' and 'upper' CLUT can be appropriately swapped in round-robin fashion so as to avoid copying data from one to the next, but that essentially takes away the option of making incremental rather than wholesale changes to the 'next' CLUT.)

The output of the CLUT's section 230 flows through a programmable interpolator 250. The interpolator 250 may be programmably enabled or disabled such that it provides low-resolution to high-resolution image enhancement in one, or both, or none of the horizontal screen direction and the vertical screen direction. The enable/disable of the interpolator's vertical and horizontal image enhancement functions may be controlled on a pixel-by-pixel (PP) basis in response to each pixel's D-bit as will be explained below.

The output of interpolator 250 flows through a set of programmably selectable or bypassable filters 260. In one mode, the filters 260 provide 1-14-1 horizontal tap filtration. In another selectable mode, the filters provide 1-6-1 horizontal tap filtration. In a third programmably selectable mode, the horizontal filtering function of section 260 is bypassed. A pre-selected one of filters 260 may be enabled or bypassed on a pixel-by-pixel (PP) basis in response to each pixel's D-bit. Pre-selection of one of filters 260 may be carried out on a per-scanline (PS) basis.

The output of filters section 260 next flows through a dithering section 270. Dithering is programmably ordered according to the X and Y coordinates of the corresponding screen pixel or it is random. Dithering may be turned on or off on a pixel-by-pixel (PP) basis in response to each pixel's D-bit.

The output of the dithering section 270 flows through a programmably bypassable conversion matrix 162. Conversion matrix 162 may be used for converting RGB-formatted image data into YUV-formatted image data. (Other conversion functions for bypassable conversion matrix 162 are also contemplated, such as YUV-to-RGB conversion.)

The digital encoder 160 adds further signals such as NTSC or PAL burst information to the digital image information and provides appropriate cross-modulation or mixing as needed for the corresponding output. The conversion matrix 162 may be enabled or bypassed on a pixel-by-pixel (PP) basis in response to each pixel's D-bit.

The corresponding analog output from the digital-to-analog converter (DAC's) section 170 may be in the form of a one-wire composite signal (Comp), or a two-wire luminance/chromiance pair (Lum/Chrom) or a three-wire RGB triad. These signals are output to the corresponding monitor 180 for display. The output signals 276 of dithering section 270 and next-described control signals 246 are provided to the exterior of the I-EARS chip for use by an external digital encoder where desired.

A variety of timing and control signals 246 pass between the digital encoder 160 and the VPP 150. Included in the signals 246 are Hsync pulses and Vsync pulses, both generated by the digital encoder 160 in synchronism with the vidCLK.

As earlier mentioned, an upstream portion 205 of the VPP 150 operates in synchronism with the sysCLK while the digital encoder 160 and a downstream portion of the VPP 150 (from the output of splittable FIFO 220 and down) operate in synchronism with the vidCLK. For odd field numbers, the corresponding Vsync pulse is generated by the DENC 160 so as to coincide in time with the corresponding Hsync pulse of the first raster line. For even numbered fields, the Vsync pulse is generated to occur approximately in the middle of the first raster line. The VPP 150 determines whether the current field is odd or even by testing for coincidence

between the Hsync and Vsync pulses and acts accordingly.

Figs. 3A and 3B combine respectively one above the next to show how differently-formatted image data may be stored in system memory and thereafter downloaded into, and re-organized within the FIFO 220.

Column 340 of Fig. 3A lists three different image formats which may be selected on a line-by-line basis for processing by VPP 150. They are: (a) Opera_LR16 format; (b) M2_16-bit format; and (c) M2_32-bit format. Although these three specific formats are shown, it is within the scope of the invention to accommodate other image-defining formats.

The memory interface circuit 205 of Fig. 3B responds to a format code stored in an FB_FORMAT register 214 (Fig. 3B) and accordingly fetches the image-defining data from system memory 130-131, re-organizes it and stores the re-organized data into appropriate positions within the splittable FIFO 220.

The splittable FIFO 220, in one embodiment, is implemented as a combination of a static random access memory (SRAM) and an access-controlling state machine. The output of this combination behaves as a single or dual set of first-in/first-out buffers (FIFO's) depending on whether the current FIFO mode is single mode or split mode. The input to this SRAM/state-machine combination functions more like a random access memory with the point or points of next-input write being controlled by the state machine and the FB_FORMAT register 214 and other control fields (LV and UV).

FB_FORMAT register 214 may be programmed to store any one of the following format-specifying codes: Opera_LR16 format, M2_16-bit format, and M2_32-bit format. The system may be expanded to handle other formats as well. The format-specifying code within the

FB_FORMAT register 214 may be modified on a per-scanline (PS) basis in synchronism with the Hsync pulse.

If a corresponding load-shadow signal (FB_FORMAT_L) is set true (to logic "1") then a corresponding, next-to-be used format-specifying code is conditionally loaded into an FB_FORMAT shadow register 212 in synchronism with the sysCLK 117. This next-to-be used code is downloaded from system memory 130-131 into the FB_FORMAT shadow register 212 by way of path 211 from the memory interface circuit during a shadow-download time period. The shadow-download period may occur any time prior to a periodically repeated transfer (213) of the shadow control data (212) into the current control register (214). The shadow-download period may occur during the horizontal active-scan period (H-SCAN). During each horizontal-blanking period (H-BLANK), the contents of the FB_FORMAT shadow register 212 are moved by way of path 213 into the FB_FORMAT register 214.

The format-specifying code for each frame buffer line is defined by a control word (VDLCW, described below) which is first stored in one of the control lists (VDL's) 142/143 of the system memory (Fig. 2) and then downloads by way of memory interface circuit 205 into the FB_FORMAT shadow register 212. The FB_FORMAT shadow register 212 is one of plural shadow-control registers that may be downloaded with shadow control data prior to transferring such shadow control data to a corresponding current-control register. Shadow-download occurs in synchronism with the sysCLK 117. Shadow-to-current copy occurs in synchronism with the vidCLK 167.

Registers 212 and 214 are just one example of a 'shadow-current' control register pair. As explained,

when a corresponding shadow-copy enabling bit (e.g., FB_FORMAT_L) or an enabling bit-string is appropriately set, the contents of the shadow register (212) are changed and thereafter copied into the current control register (214) at the appropriate time (e.g. when a new VDL takes effect in approximate coincidence with or just before the Hsync pulse).

Another example of a 'shadow-current' control register pair is shown schematically at the bottom of Fig. 3B at 241 as a compounded set of boxes. Yet another example of a 'shadow-current' control register pair is shown at 242. Each of the respective current control registers in the HPD and VPD 'shadow-current' control register pairs, 241 and 242, is independently modifiable on a per-scanline basis (PS basis) as is the FB_FORMAT register 214.

Referring to Fig. 3A, image data is periodically fetched by the memory interface circuit 205 from the system memory 130/131 in the form of 32-bit wide memory words. In the so-called Opera_LR16 format, each 32-bit memory word contains 16 bits defining a first pixel of a first frame buffer line and 16 further bits defining a vertically adjacent pixel of a second frame buffer line.

As earlier explained, 15 of the bits in each memory halfword may be used for defining respective 5-bit wide color components such as R-G-B or Y-U-V. (If the system memory data represents RGB color components and the DENC 160 expects YUV, then it is generally desirable to enable the downstream RGB-to-YUV conversion matrix 162. If the system memory data already represents YUV color components as expected by the downstream DENC 160, then it is generally desirable to bypass the downstream conversion matrix 162.)

The 16th bit of each halfword is a so-called 'D-bit' which may be used for modulating various parameters of the I-EARS 150-160-170 on a pixel-by-pixel basis, including enablement or bypass of the downstream conversion matrix 162.

In the Opera_LR16 format, a first halfword of each fetched 32-bit wide memory word represents the D-RGB information (or alternatively, the D-YUV information; or further alternatively, the D-ABC information, where A,B,C are coordinates in another 3-dimensional color space) for a first frame buffer pixel on a first frame buffer line. A second halfword in the same 32-bit memory word represents the D-RGB information (or D-YUV or D-ABC information) for a second pixel of the frame buffer that is vertically adjacent within the frame buffer to the first pixel. (A 'frame-buffer line' is not necessarily the same as a display line on the screen of the VDU 180. If vertical pixel-doubling (VPD) is active, two screen display lines are generated for each corresponding frame-buffer line. If horizontal pixel-doubling (HPD) is active, two horizontally-adjacent screen display pixels are generated for each corresponding frame-buffer pixel. This will be detailed below.)

Reference numeral 301 of Fig. 3A points to a first 32-bit memory word that is formatted in accordance with Opera_LR16 format. Reference numeral 302 points to a second 32-bit memory word of the same frame-buffer band. The second memory word 302 contains the D-RGB information (or D-YUV or D-ABC information) for, by way of example, pixel number N of frame buffer line-0 and also for pixel number N of frame buffer line-1 while the first memory word 301 contains the corresponding information for preceding pixels of the respective frame buffer lines, 0 and 1. The same

organization repeats for frame-buffer lines 2 and 3 if any, 4 and 5, and so forth.

The 32-bit wide, Opera_LR16 formatted pixel words are arranged sequentially in the memory space of system memory 130-131 for each successive, vertically-adjacent pair of pixels of frame buffer lines 0 and 1, then for each successive pixel pairs of frame buffer lines 2 and 3 if any, and so forth.

The system software defines which of the two halfwords in an Opera LR16 memory word 301 belongs to a frame buffer 'upper-line' (UL) and which belongs to a frame buffer 'lower-line' (LL). Software definition of UL and LL occurs as follows: The system CPU 110 writes a 'forced video address' into a DMA control register (210) within the VPP 150. The forced video address (FV0A for even fields, FV1A for odd fields as will be detailed below) points back into system memory into a video display control list (VDL). There are two 'mandatory' entries in each VDL which are referred to respectively as the first lower-line in-current frame-buffer address (LL_FB address) and the first upper-line in-current frame-buffer address (UL_FB address).

The LL_FB and UL_FB address words are each at least 32-bits wide and each can point to any group of bits within the system memory 130-131 beginning at a byte boundary. (A byte contains 8 sequential bits of data and may also contain additional error-detect/correct bits such as a parity bit). The VPP 150 fetches the pointed-to LL_FB and UL_FB address words, when such a fetch is called for, but ignores the least significant bit of each of the LL_FB and UL_FB address words. Thus the VPP 150 has an access resolution into system memory down to the halfword level. The LL_FB address word points, as shown in Fig. 3A near 301, to the memory halfword that is to be directed to a lower-

line (LL) section 221 of the splittable FIFO 220. The UL_FB address word similarly points to the halfword in memory that is to be directed to the upper-line (UL) portion 222 of the splittable FIFO 220.

Referring to Fig. 3B, it is seen that the D-RGB information (or D-YUV or D-ABC information) of the line-1 halfwords are directed by the memory interface circuit 205 into the lower-line FIFO section 221. The D-RGB information (or D-YUV or D-ABC information) of the line-0 halfwords are similarly routed by memory interface circuit 205 into the upper-line FIFO section 222.

Unlike the 32-bit per-word format of the system memory 130-131, the entries of FIFO 220 have a 25-bit format. For the split mode, each side 221 and 222 of splittable FIFO 220 is preferably at least 64 entries deep. In the unsplit mode, splittable FIFO 220 can operate as a single FIFO having a depth of at least 128 entries, each entry being 25-bits wide. The depth of the splittable FIFO 220 can of course, be changed as appropriate to account for memory access latency within the system 100.

One bit of each FIFO entry stores the D-bit of the corresponding memory halfword and the remaining 24 bits are subdivided into three sections (e.g., each 8-bits wide) for storing the respective color space coordinate information (RGB or YUV or the generic, ABC) of the corresponding frame buffer pixel.

In the case of Opera LR16 format, where there are only 5 bits in system memory for each color space coordinate, the least significant three bits of each 8-bit wide one of the R, G, and B fields (or A,B,C fields) within the FIFO 220 are filled with zeroes. The more significant five bits are filled with the

corresponding five bits from the respective color space coordinate fields of the corresponding memory halfword.

Referring to memory words 316 and 317 of Fig. 3A, these show how image data may be formatted according to the so-called 'M2_16-bit' format. Each 32-bit wide memory word stores the D-RGB (or D-ABC) information of two pixels that are horizontally adjacent within the system-memory frame-buffer. When the FIFO 220 is in split mode, the UL_FB address word points to the memory start location of 'upper-line' pixels and the LL_FB address word points to the start of 'lower-line' pixel data. FIFO 220 is filled by alternately fetching blocks of memory words corresponding to upper and lower frame buffer lines. When the FIFO 220 is in the unsplit mode, only the LL_FB address word is used for data fetch.

When a software-defined number of frame-buffer pixels per frame-buffer line is exhausted, the UL_FB and/or LL_FB address words are appropriately incremented to point to the respective next locations in system memory that contain the logically-next set of upper and lower frame-buffer lines. (See below explanation of the MOD field in the Frame-buffer DMA Control Word.)

Fetches of data from system memory are made in quantums of or quadbytes (32 bytes each). The memory interface circuit 205 tries to minimize data bus utilization time (on D0 or D1) by fetching as many quantum fetches as possible per FIFO fill.

FIFO 220 has two levels associated therewith for preventing overfill and underfill. Level 314 (Fig. 3B) is the near-empty level and is programmably-adjustable (in one embodiment to FIFO filled/full ratios of 32/128, 64/128 and 96/128). The upper level 312 is the ready-for-next-fetch level, also referred to as the

overfill-protect level 312. Level 312 is preferably set so that one or more memory quadbyte fetches may be made without overfilling the FIFO. In one embodiment, the FIFO filled/full ratio for the overfill-protect level 312 is 96/128.

If the filled level of the FIFO 220 is or drops into the range between levels 312 and 314, then fetch requests for filling the FIFO are made at normal priority. The requested fetch size is equal to the difference between the full level and the overfill-protect level 312. (In one embodiment, the requested fetch size is equal to 32/128ths of the FIFO full depth.)

When the amount of stored data within the FIFO 220 is or falls below the programmably-definable near-empty level 314, fetch requests are automatically generated at high priority. The requested fetch size is still equal to the difference between the full level and the overfill-protect level 312 but of course repeats at high priority until the FIFO filled level rises above the near-empty level 314. The FIFO 220 accordingly requests memory access at normal and high priority in a manner which automatically prevents overfill and automatically minimizes the likelihood of underfill.

Referring to the memory words labeled as 332 and 333 in Fig. 3A, these are formatted according to a so-called M2_32-bit format. Each of memory words 332 and 333 stores the D-RGB (or D-ABC) information of a single high resolution pixel. Each of the R,G,B fields (or A,B,C fields) of each M2_32-bit formatted memory word are respectively 8-bits wide. The D-bit field is one bit wide. A 7-bit wide reserve field (X) is further included in each of memory words 332 and 333. The D-RGB (or D-ABC) information of each of memory words

332 and 333 is routed by the memory interface circuit 205 into a respective 25-bit wide FIFO entry location.

Referring to Fig. 3B, the splittable FIFO 220 has a FIFO output circuit 327 that is clocked differently depending on the 'current' states of two pixel-rate control register sets, the horizontal pixel-doubling (HPD) register set 241 and the vertical pixel-doubling (VPD) register set 242. Each of register sets 241 and 242 is shadowed and update-able on a per-scan-line (PS) basis.

Each of the horizontal pixel-doubling (HPD) and vertical pixel-doubling (VPD) functions may be independently turned on or off. When pixel-doubling is enabled in a given direction (vertical or horizontal), interpolation may be turned on or off in that given direction on a pixel-by-pixel basis. If HPD is turned on (active), then two screen pixels are generated in the horizontal direction for each frame-buffer pixel. If horizontal interpolation is turned off at the same time, the two horizontally-adjacent screen pixels will have the same color values. If horizontal interpolation is instead turned on at the same time, the color components of each of the two screen pixels will be set according to sub-position weighted interpolation with neighboring frame buffer pixels.

Similarly, when VPD is turned on (active), two screen pixels will be generated in the vertical direction for every frame buffer pixel. If vertical interpolation is turned off, the two vertically-adjacent screen pixels will have identical color values. If vertical interpolation is turned on, the color value of each of the two vertically-adjacent screen pixels will be defined according to weighted interpolation of the current frame-buffer pixel with neighboring frame-buffer pixels.

If pixel-doubling is turned off in a given direction (horizontal or vertical), then it is not advisable to turn on interpolation in that same direction. The allowed combinations are shown in the below TABLE-1.

TABLE-1

Pixel Doubling is active in the indicated direction?	Interpolate in the indicated direction?
Yes	No
Yes	Yes
No	No
No-(This is a disallowed mode)	Yes-(This is a disallowed mode)

The bottom of Fig. 3B indicates the rates at which the FIFO output circuit 327 is clocked for different settings of HPD and VPD. If HPD is on and VPD is off (HV=0,1), then the FIFO output circuit 327 is clocked at twice the video pixel rate (vidPRx2). For the pixel-doubling combinations of HV=0,0 and HV=1,1, the FIFO output circuit 327 is clocked at a frequency equal to the video pixel rate (vidPRx1). For the pixel-doubling combination of HV=1,0, the FIFO output circuit 327 is clocked at half the pixel frequency of the video monitor (vidPRx1/2). Multiplexer 328 receives the respective clock signals, vidPRx2, vidPRx1, and vidPRx1/2 and supplies the appropriate clock signal to the FIFO output circuit 327 in accordance with the current settings of HPD register set 241 and VPD register set 242. The 25-bit-wide outputs of upper line FIFO 222 and lower line FIFO 221 next flow to the CLUT section 230 at the rate selected by multiplexer 328.

Referring to Fig. 4A, the five more significant bits of each (5+3)-bits wide, R or G or B color component field output by splittable FIFO 220 is routed to a respective 5-bit-wide address input of a corresponding R or G or B subCLUT. Each (5+3)-bits wide, color component field output by the splittable FIFO 220 is referred to as a 'PEN' signal. The designations of R or G or B for each PEN signal may be substituted by a respective Y or U or V; or more generically by a respective A or B or C.

Fig. 4A shows how the lower line CLUT 231 is subdivided into its respective R, G and B subCLUT's 431, 432 and 433. Each of subCLUT's 431-433 stores thirty-two 8-bit wide entries. Each of subCLUT's 431-433 is independently addressable for purposes of reading from or writing to its respective thirty-two entries. Additionally, there is a 24-bit wide 33rd storage entry which is alternatively selected as the output of the lower-line CLUT 231 when a zero-detector circuit 430 detects the condition where the five more significant bits of each of the RGB PEN output fields from the LL-FIFO 221 are simultaneously equal to zero. Zero detector 430 may be implemented as a 15-input NOR gate.

The least significant of the 5 address input bits to the B subCLUT 433 may be selected from four sources on a pixel-by-pixel basis. It can either be the fifth most significant bit (5th msb) of the corresponding 8-bit wide LL-FIFO output for the B/V PEN field or the 5th msb of the 8-bit wide G/U PEN field of LL-FIFO 221, or a zero ("0") or a one ("1"). The selection is made by multiplexer 434. Degradation of color-palette resolution along the blue axis of the RGB color space is believed to be less noticeable to the human eye than similar loss along the red and green axes.

The selection control input of multiplexer 434 is defined by the output of multiplexer 435. Multiplexer 435 has as its inputs a first 2-bit wide blue-select register, BS_0 and a second 2-bit wide blue-select register, BS_1. Register BS_0 forms part of a below-described VDC0 register. Register BS_1 forms a part of a below-described VDC1 register. Each of the VDC0 and VDC1 registers is modifiable on a per-scanline (PS) basis from a VDC0/1 shadow register.

The D-bit output from LL-FIFO 221 can change on a per-pixel basis (PP) given that the D-bit value for each frame-buffer pixel is established in the system memory 130-131.

If the contents of the BS_0 and BS_1 registers are different, then the LL-FIFO output D-bit will modulate the output of multiplexer 435 accordingly. If the contents of current control registers BS_0 and BS_1 are the same, then the setting of the D-bit will have no effect on the output of multiplexer 435. It will be fixed to the common control code stored in BS_0 and BS_1.

Multiplexer 434 is included in the VPP so that the 5th msb of the B/V PEN signal can be optionally used for independent pixel-by-pixel control of one or both of so-called H and V subposition bits used by the downstream interpolator 250. If the 5th msb of the B/V PEN signal is used for modulating the H and V subposition bits on a pixel-by-pixel basis for interpolation purposes, such modulation of this bit may cause undesired artifacts in the rendered image if the same bit is also used for defining the least significant address bit entering the B subCLUT 433. It may be desirable to instead fix the least significant address input bit of the B subCLUT 433 to a zero ("0") or to a one ("1") or to make it a copy of the least significant address bit going into the G subCLUT 432.

Multiplexer 434 provides these options in response to the output of the B-select control multiplexer 435 as indicated in Fig. 4A.

Lower-line CLUT 231 provides a software-defined mapping of each more significant 5-bits of the LL-FIFO PEN signals into a corresponding 8-bit wide signal. The 8-bit wide output of the R subCLUT 431 next flows to a bypass multiplexer 441. In a first state, the bypass multiplexer 441 outputs the same 8-bit wide color component signal received from the R subCLUT 431. In a second mode, the bypass register 441 outputs a combined 8 bits (8=5+3) received from the respective 5 and 3-bit wide outputs of delay units 437 and 438. The output of delay unit 437 is simply the most significant 5 bits of the R/Y PEN signal output from the LL-FIFO 221, but delayed in time by an amount corresponding to the read-mode propagation delay of the R subCLUT 431. The 3-bit wide output of delay unit 438 is similarly delayed so as to coincide in time with the output of delay unit 437. Delay unit 438 receives a selected one of four 3-bit wide signals from a CLUT-bypass drive multiplexer 436.

The selection control input of the CLUT-bypass drive multiplexer 436 is controlled by a 2-bit wide CBPSEL register 443 (CLUT-bypass select register 443). One of the four inputs to the CLUT-bypass drive multiplexer 436 is derived from the corresponding three least significant bits (lsb's) of the 8-bit wide R/Y PEN signal output from FIFO 220. A second of the inputs to multiplexer 436 is derived from the corresponding three most significant bits (msb's) of the R/Y PEN signal output from FIFO 220. A third of the inputs of multiplexer 436 is a predefined constant 3-bit field such as 000. The fourth input to multiplexer 436 can be any other 3-bit wide field

including a pseudo randomly generated field. In the embodiment further described below, the fourth input is a repeat of the constant 000 field.

The CBPSEL register 443 forms a part of a VDLI register described below. The VDLI register is modifiable on a per-scanline (PS) basis.

Another register within the VDLI register is the illustrated CLUT-bypass enable register (CLUTBP) 445. The CLUTBP register 445 drives one input of AND gate 442. A second input of AND gate 442 is driven by the corresponding D-bit output from FIFO 220. When the CLUTBP register 445 outputs a logic "0", the CLUT bypass function is disabled and multiplexer 441 simply passes through the 8 bits from the R subCLUT 431. When the output of CLUTBP register 445 is logic "1", CLUT bypass is enabled and the D-bit passes through AND gate 442 to control the selection terminal of bypass multiplexer 441.

When CLUTBP=1, the D-bit can be used to increase the color palette space available to programmers by as much as a factor of two. One half of the doubled palette space may be defined by the configuration of the CLUT bypass circuitry formed by elements 436-437-438 and 443. The other half of the doubled palette space may be defined by the color-mapping entries downloaded into the programmable LL-CLUT 231. Of course, if some or all of the downloaded color-mapping entries are redundant with the output of the CLUT bypass circuitry formed by elements 436, 437, 438, 443; then the color palette space available to programmers will be accordingly reduced.

Although Fig. 4A shows the CLUT bypass circuitry only for the R subCLUT 431, it should be understood that similar circuitry is provided for bypassing the G subCLUT 432 and further similar circuitry is provided

included in that embodiment of the VPP 150.

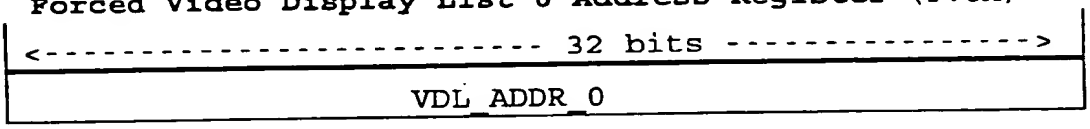
Each listed register is 32-bits wide. Each register is listed according to its full name and also according to an identifying acronym. TABLE-2 also indicates which registers that are readable by the CPU and/or writable into by the CPU and/or clearable by the CPU. If the entry is "R" only, that means that the register is only readable by the CPU but cannot be written to by the CPU. TABLE-2 also indicates which registers may be written to during a DMA fetch of video display list control words and which registers are read by the DMA mechanism. TABLE-2 also indicates the coupling between the digital encoder and the VPP control registers. The indicator "(W)" means that the digital encoder indirectly defines the contents of this register by means of the horizontal and vertical video sync pulses (Hsync and Vsync).

TABLE-2

Register Name	Acronym	Read/Write/Clear by CPU	Read/Write by DMA	Read/Write by DENC
Forced VDL 0 Address	FV0A	R/W	R	
Forced VDL 1 Address	FV1A	R/W	R	
Active Video Display Info	AVDI	R	W	
Video Display List Info	VDLI	R	W	
Video Display Control 0	VDC0	R	W	
Video Display Control 1	VDC1	R	W	
Video Display Screen Info	VDSI	R/W		
Video Display Location	VLOC	R	R	(W)
Vertical Line Interrupt	VINT	R/W		(W)
Vert. Line Interrupt Clear	VINT	C		
Dither Matrix 0	DMT0	R/W		
Dither Matrix 1	DMT1	R/W		
LFSR Seed	LFSR	R/W		
Digital Video Encoder	DVER	R/W		R
DMA Snoop Control	DSNP	R/W	R	

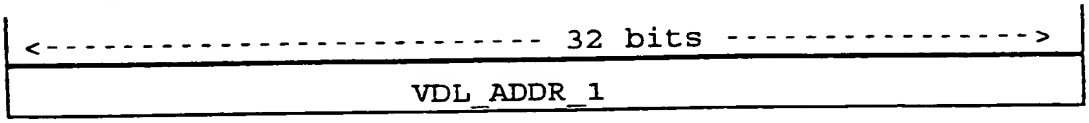
The fields in the specific registers of TABLE-2 are now detailed as follows.

Forced Video Display List 0 Address Register (FV0A)



VDL_ADDR_0 is the forced start address for the active video display list (VDL) when the current screen is an even field.

Forced Video Display List 1 Address Register (FV1A)



VDL_ADDR_1 is the forced start address for the active video display list (VDL) when the current screen is an odd field.

Active Video Display Info Register - (AVDI)

11 bits	11 bits	1 bit	1 bit	8 bits
HSTART	HEND	HPD	VPD	--- (reserved)

The contents of the Active Video Display Info Register (AVDI) may be defined by an Active Video Display Control Word (AVDCW) that is optionally included in the video display list (VDL). The CPU 110 can only read these fields:

HSTART (11 bits):

The digital encoder (160) can be instructed to keep horizontal blanking on beyond the end of the front porch in the output NTSC or PAL video signal. HSTART defines for the digital encoder the number of screen pixels after the front porch for which horizontal blanking is to be kept active.

HEND (11 bits):

This horizontal-end count defines the number of vidCLK pulses or the number of screen pixels for which horizontal blanking is not active. In other words, this defines the effective horizontal screen width for a corresponding line in the frame-buffer. At the end of the HEND count, horizontal blanking is turned back on again irrespective of whether the ending pixels of the corresponding frame-buffer line have not yet been exhausted.

HPD (1 bit):

Horizontal pixel doubling active. A logic "1" in this bit position means that horizontal pixel doubling is turned on so as to produce two active horizontal video pixels for every frame buffer pixel.

VPD (1 bit):

Vertical pixel doubling is active. A logic "1" here indicates that vertical pixel doubling is turned on such that two active vertical video pixels are produced for every frame buffer pixel.

Video Display List Info Register - VDLI

1	1	2	2	1	1	1	1	1	11
CLUT_BP	RGB	CBP_SEL	FB_FORMAT	FILT_TYPE	FTRAN	BKG_TRAN	VIL	RDM	---

The contents of the Video Display List Info Register (VDLI) may be defined by a Video Display List Control Word (VDLCW) found within the VDL. The VDLI fields are as follows:

CLUTBP (1 bit):

A logic "1" here indicates that CLUT bypass is enabled.

RGB (1 bit):

A logic "1" here indicates that the original image is in RGB format and the RGB-to-YUV conversion matrix 162 in the digital encoder should be turned on. A logic "0" at this bit position indicates that the original image is already in YUV format and the matrix converter 162 is to be bypassed.

CBPSEL (2 bits):

CLUT bypass select. When either of the soft CLUT's 231-232 is bypassed, the three least significant of the 8 bits output by multiplexer 441 (Fig. 4A) are supplied from a selected on of

four sources. The below TABLE-3 shows one possible coding for the four sources.

TABLE-3

CBPSEL (2 bits)	Meaning
0	Fill with 0's
1	Fill with 3 msb's from FIFO
2	Fill with 0's
3	Fill with 3 lsb's from FIFO

FBFORMAT (2 bits):

Frame buffer format. As explained above, there are three image encoding formats for the frame buffer within the system memory. The below TABLE-4 shows the respective formats. A fourth format may be added if desired.

TABLE-4

FBFORMAT (2 bits)	Meaning
0	16-bit Opera LR format
1	M2 16-bit format
2	M2 32-bit format
3	reserved

FILT_TYPE (1 bit):

Filter type. If post-interpolation filtering is enabled, this field selects one of two filter types: full resolution (=1) or quarter resolution (=0).

FTRAN (1 bit):

Force transparency. If this bit is set, a transparency signal is sent to the digital encoder and is forced to be on for the entire

line.

BKGTRAN (1 bit):

Background transparency. If this bit is set, the transparency signal sent to the digital encoder will be set whenever a background pixel is detected. The zero-detector 430 of Fig. 4A detects the so-called background pixel condition.

VIL (1 bit):

Vertical interpolation line. If this bit is set, vertical interpolation for the first scan line associated with this VDL will be turned off. The remaining lines associated with this VDL will have vertical interpolation enabled or not depending on the D-bit and the VIE_0/1 set in registers VDC0 and VDC1.

RDM (1 bit):

Random dither matrix index. If this bit is set, an LFSR seed will be used to randomly select dither matrix entries for use with each pixel. If this bit is clear, the row and column position values of the pixel within the active display area will be used to index into the matrix and use the corresponding entry for that pixel.

Video Display Control 0 Register - VDC0

2	2	2	1	1	1	1	1	11
HS	VS	BS	HIE	VIE	FE	DE	MBE	---

There are two video display control registers, VDC0 and VDC1. The contents of the VDC0 and VDC1 registers are updatable on a per-scanline basis (PS basis) by means of a below-described video display control word. The D-bit selects all the bits of one of registers VDC0 and VDC1 on a per pixel basis as the active set. A portion of this multiplexing scheme is shown in Fig. 4A by element 435 and the BS_0 and BS_1

registers. If a control parameter is to remain unaffected by the D-bit, then the corresponding fields in the VDC0 and VDC1 registers should be set to the same value. If they are different, the D-bit will select one of the different values as the active value for each corresponding pixel.

HS (2 bits):

Horizontal subposition bit select. This is used in Opera-style interpolation. The below TABLE-5 shows the encoding for four possible sources.

TABLE-5

HS	Meaning
0	Use D-bit
1	Use Blue Pen 5th msb
2	Set to 0
3	Set to 1

VS (2 bits):

Vertical subposition bit select. The vertical subposition bit is similarly selectable from four sources as shown by the below TABLE-6.

TABLE-6

VS	Meaning
0	Use D-bit
1	Use Blue Pen 5th msb
2	Set to 0
3	Set to 1

BS (2 bits):

Blue PEN 5th msb select. As seen in Fig. 4A, the fifth most significant bit of the 8-bit wide blue

PEN signal can be redefined on a per pixel basis. The following TABLE-7 shows the options.

TABLE-7

BS	Meaning
0	Use 5th msb of B/V PEN signal
1	Use Green PEN 5th msb
2	Set to 0
3	Set to 1

HIE (1 bit):

Horizontal interpolation enable. As explained above, when horizontal interpolation is active, the interpolator 250 outputs two horizontal pixels for each frame buffer pixel where the values of the two horizontal pixels are defined separately by an interpolation function. When HIE is turned off, the two horizontal pixel values that are output are the same and are equal to that of the corresponding low-resolution input pixel then being processed by the interpolator 250.

FE (1 bit):

Filter enable. When reset to logic "0", the post-interpolation filtration 260 will be bypassed. When set to logic "1", the filter function defined by FILT_TYPE in the VDLI register is enabled.

DE (1 bit):

Dither enable. When this bit is set, the dither function 270 is enabled. When this bit is 0, the dither function is bypassed.

MBE (1 bit):

Matrix bypass for digital encoder. When set to 0, disables the RGB-to-YUV conversion matrix 162 within the digital encoder 160.

Video Display Control 1 Register - VDC1

2	2	2	1	1	1	1	1	11
HS	VS	BS	HIE	VIE	FE	DE	MBE	---

See field definitions for VDC0.

Programmers may use the per-pixel modulation capabilities of the D-bit in combination with the settings of the fields in video display control registers VDC0 and VDC1 to create a number of interesting screen effects.

For example, if horizontal and/or vertical interpolation (HIE and VIE) is turned off in a screen region defining an object border such as the border of a score box, then the same color value will be replicated in pairs of adjacent pixels. This can be used to create a visual effect of strong continuity from one pixel to the next. Such strong continuity may be desired for drawing a sharp outline around a visual object such as a score-keeping box in a video game. When horizontal and/or vertical interpolation are turned on, the difference between adjacent pixels tends to be more smooth rather than sharp. This smoothing effect may be turned on and off as desired to create visual effects such as that of a rounded or foggy object edge.

Additional control over the smoothness or sharpness of horizontally adjacent pixels may be obtained by modulating the FE (filter enable) bit.

In some regions of a displayed object it may be desirable to turn on the dithering function 270, so as

to, for example, stitch together the seams of frame buffer objects that were obtained from different sources. In other areas of the same displayed image, it may be disadvantageous to have dithering turned on. For example, if a certain region is to have a solid same color throughout, dithering would undesirably add visual noise to this certain region. In this situation, it is beneficial to turn the dithering function off. The pixel-by-pixel control over the D-bit lets programmers turn dithering on and off as desired.

The matrix-enable/bypass control bit (MBE) may be similarly used to define window regions within the screen display for which RGB-to-YUV conversion is or is not desired. For example, if a given region in the system-memory frame-buffer is already encoded as YUV (because, for example, it is the output of an MPEG decompression circuit that writes into system memory), then it will be generally undesirable to apply the RGB-to-YUV transformation to such a region. On the other hand, if an adjacent portion of the same frame-buffer is RGB encoded, it will be generally desirable to apply the RGB-to-YUV transformation to this adjacent region. The sizes and locations of YUV and RGB pre-encoded regions can vary. The pixel-by-pixel control over the D-bit lets programmers turn conversion on and off as desired.

Video Display Screen Info Register - VDSI

11	7	11	3
HRES	FRATE	VRES	---

HRES (11 bits): Horizontal resolution. This is used as a CPU general storage area by which various software modules can define to

each other or determine the type of VPP processes being used.

VRES (11 bits): Vertical resolution.

FRATE (7 bits): Frame rate. Another CPU general-usage storage area.

Video Display Location Register - VLOC

11	1	11	9
HCOUNT	FIELD	VCOUNT	---

HCOUNT (11 bits):

Horizontal pixel count after HSYNC (dve_vidHSOn). Not implemented.

FIELD (1 bit):

Field number. Indicates the odd/even nature of the current field that is being displayed. The field is odd (=1) if the HSYNC and VSYNC pulses are coincident. If the VSYNC pulse occurs in the middle of a line, the new field is even. (=0)

VCOUNT (11 bits):

Vertical line count after VSYNC (dve_vidVSOn). The vertical line number after the start of a new field that is being displayed. Due to the asynchronous nature of the video clock, this value may be off from the actual line value by 1.

Vertical Line Interrupt Register - VINT

1	11	1	11	8
VINT0	VLINE0	VINT1	VLINE1	---

VINT0 (1 bit):

Vertical Line Interrupt 0. This bit will get set when the video display has reached the vertical line number set in VLINE0. This bit being set

will cause the vid_busVertint0 signal to be asserted for interrupting the CPU. To clear this bit, the CPU needs to write to the VINT-clear register with this bit set to 1.

VLINE0 (11 bits):

Vertical Line Number 0. One of the two vertical line numbers where the video display unit should cause an interrupt. When the display reaches this line number, the VINT0 bit will be set.

VINT1 (1 bit):

Vertical Line Interrupt 1. This bit will get set when the video display has reached the vertical line number set in VLINE1. This bit being set will cause the vid_busVertInt1 signal to be asserted. To clear this bit, the CPU needs to write to VINT-clear register with this bit set to 1.

VLINE1 (11 bits):

Vertical Line Number 1. One of the two vertical line numbers where the video display unit should cause an interrupt. When the display reaches this line number, the VINT1 bit will be set.

Dither Matrix 0 - DMT0

4	4	4	4	4	4	4	4
MT00	MT01	MT02	MT03	MT10	MT11	MT12	MT13

Dither Matrix 1 - DMT1

4	4	4	4	4	4	4	4
MT20	MT21	MT22	MT23	MT30	MT31	MT32	MT33

These two registers (DMT0 and DMT1) store a set of dither matrix values that are downloaded into a 4x4 dither matrix within dithering section 270. Each entry in the 4x4 matrix is a signed 4-bit value (in two's

complement form). If dithering is enabled, a selected one of the signed values from the matrix will be added to each color component and wrap-around clipping will be performed. In ordered dithering the matrix entry to be used is determined by the less significant bits of the row and column number of the pixel being displayed. In random dithering the matrix entry to be used is picked pseudo-randomly. The DMT0 register contains the MT0x and MT1x values, while the DMT1 register contains the MT2x and MT3x values. Their placement within the 4x4 matrix is shown in below TABLE-8.

TABLE-8

MT00	MT01	MT02	MT03
MT10	MT11	MT12	MT13
MT20	MT21	MT22	MT23
MT30	MT31	MT32	MT33

LFSR Seed Register - LFSR

32
LFSRSEED

LFSRSEED (32 bits):

Linear feedback shift register seed. When this register is written to, the contents are loaded into the LFSR within the dithering unit and used for pseudo-random number generation.

DMA Snoop Control - DSNP

1	1	2	28
FSN	VSN	FIFHP	---

FSN (1 bit):

Frame Buffer FIFO DMA Snoop enable.

VSN (1 bit):

VDL DMA Snoop enable.

FIFHP (2 bits):

FIFO high priority mark. This field determines when the FIFO is deemed near-empty and will change its memory request to high priority. The below TABLE-9 shows the respective settings.

TABLE-9

FIFHP	High priority level
0	if FIFO holds below 32 entries
1	if FIFO holds below 64 entries
2	if FIFO holds below 96 entries
3	Always

The Active Video Display List (VDL)

The forced VDL address registers mentioned earlier, FV0A and FV1A, cause the DMA engine within the VPP to begin fetching control words from a corresponding, active video display list for respective even and odd fields.

Each active video display list (VDL) should contain a header of at least four mandatory words followed by an optional trailer. Each combination of mandatory header and optional trailer may point to another such combination or back to itself in linked-list style.

A VDL entry has a format as shown in the below TABLE-10.

TABLE-10

Frame buffer DMA control word (FB_DMA)
Frame buffer lower-line address (FB_LL)
Frame buffer upper-line address (FB_UL)
Pointer to next VDL entry (PA)
Optional CLUT-download or video control word
Optional CLUT-download or video control word
***
***
***

The first mandatory word in the VDL entry header is the frame buffer DMA control word. The structure of this control word is as follows:

Frame buffer DMA control word (FB_DMA)

8	1	1	1	1	1	6	9	4
MOD	VDE	PF	UM	LV	UV	NW	NL	---

MOD (8 bits):

Modulo value for next line address generation. If the LV bit is not set, the value contained in this field will be left-shifted 5 and added to the current lower-line data address to generate the next lower-line data address. This eighth bit field is shifted so that values from 32 to 8160 (in increments of 32) may be used as the modulo value to add per the below TABLE-11.

TABLE-11

Display type	Modulo value to add
Low resolution NTSC 32-bit M2	1280
Low res. NTSC 16-bit M2	640
Low res. NTSC 16-bit LR Opera	1280
High res. NTSC 32-bit M2	2560
High res. NTSC 16-bit M2	1280
Low res. PAL 32-bit M2	1536
Low res. PAL 16-bit M2	768
Low res. PAL 16-bit LR Opera	1536

VDE (1 bit):

Video DMA enable. This bit enables the fetching of frame buffer data into the FIFO and it indicates that the screen line is active. If VDE is logic "0", the DENC 160 keeps the horizontal-blanking (H-BLANK) for the indicated number of lines (NL). No image data is fetched into the FIFO. The active video may thus be started any desired number of lines below the Vsync position and system bus time is not wasted on non-displayed data.

PF (1 bit):

Physical address format. If this bit is set, the physical address in the fourth word of this VDL entry is a relative address. If this bit is not set, the address is an absolute address.

UM (1 bit):

Upper scanline address -use modulo. If this bit is set, --and UV is false-- the next upper scanline address will be generated by adding the modulo value to the current upper line (UL) address. If this bit is not set, --and UV is false-- the lower-line address will be used as

the next upper-line address. (In other words, the current lower scanline will become the upper scanline when the next line is being displayed. $UL_{new}=LL_{old}$.)

LV (1 bit):

Lower scanline data address validity. If this bit is set, the second word in this VDL entry will be loaded as the new lower scanline data address to fetch from. Else a modulo value is added to the current address to generate the next lower-line (LL) address.

UV (1 bit):

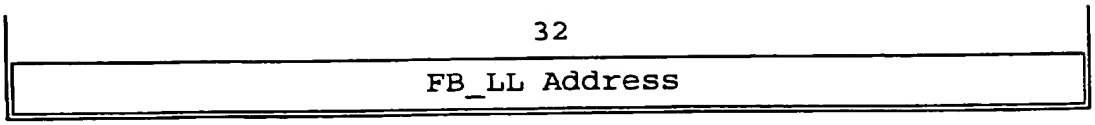
Upper scanline data address validity. If this bit is set, the third mandatory word in this VDL entry will be loaded as the new upper scanline data address to fetch from. Else the next upper scanline address will be generated per the setting of UM.

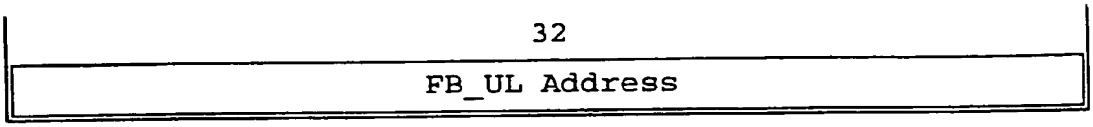
NW (6 bits):

Number of words in this VDL entry. Since an optional number of control and CLUT data words may follow the four required words in the VDL header, this field defines the total number of words that are present in this VDL DMA transfer. A value of zero is taken to mean that there are 64 system memory words in this transfer.

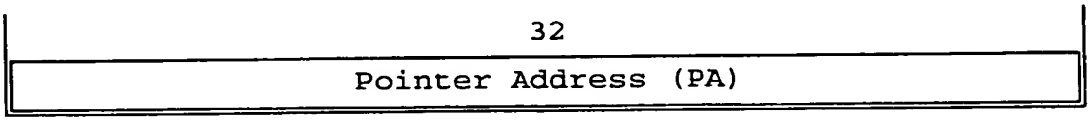
NL (9 bits):

Number of scan lines this VDL entry applies to. This value indicates the number of scanlines to wait before fetching and processing the next pointed-to VDL entry. A value of 0 means that this VDL entry applies to the remainder of this field no matter how many lines are left. The next VDL in such a case will be the forced VDL at the beginning of the next new field.



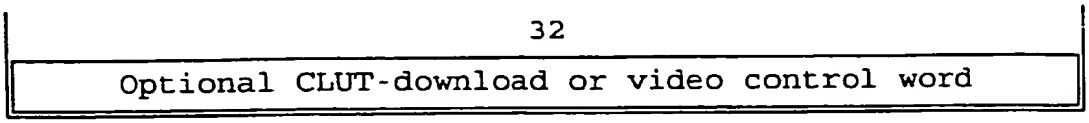


The second and third mandatory words in the VDL entry header contain 32-bit wide absolute address values respectively pointing to the locations in system memory where the first lower scanline for this VDL entry resides and where the first upper scanline for this VDL entry resides. The respective upper and lower scanline addresses are loaded into the corresponding VPP control registers only if the corresponding valid bits LV and UV in the frame buffer DMA control word are set. If not set, the VPP increments its previous values as explained above.



The fourth mandatory word in the VDL entry header is a pointer address to the next VDL entry to be fetched and processed by the VPP when the number of lines (NL) of the current VDL entry are exhausted. The pointer address (PA) can be absolute or relative depending on the setting of the pointer-format (PF) bit in the frame buffer DMA control word (FB_DMA). If the pointer address is a relative value, then the corresponding absolute address is calculated as follows:

next VDL start address = current VDL start address + 4 + PA



The optional fifth and further words of a given VDL entry can each be either a CLUT-update word or a video-display-path modifying control word. A control type code (CT) in each such optional word defines whether it is a CLUT-update word or a video-display-path modifying control word, and if so, what specific kind.

CLUT update words may be used to modify the contents of the next CLUT 235. Video-display-path modifying control words may be used to modify the various parameters along the video display path, as shall become apparent below.

There are at least five different kinds of CLUT update words as specified below. A first kind of CLUT update word writes all of its respective RGB values to the corresponding address in the next-CLUT 235. A second kind of CLUT update word writes only the contents of its BVALUE field to the corresponding address in the B sub-CLUT portion of the next-CLUT 235. A third kind of CLUT update word writes only the contents of its GVALUE field to the corresponding address in the G sub-CLUT portion of the next-CLUT 235. A fourth kind of CLUT update word writes only the contents of its RVALUE field to the corresponding address in the R sub-CLUT portion of the next-CLUT 235. A fifth kind of CLUT update word writes all three of its RGB values to a corresponding background entry in the next-CLUT 235. The general format of a CLUT update word is as follows:

CLUT Update Word

3	5	8	8	8
CT	CADDR	RVALUE	GVALUE	BVALUE

The fields in the CLUT update word are now detailed.

CT (3 bits):

This field defines the optional word type which can be a CLUT update word or a video display path control word. Up to eight different control code combinations may be used with this 3-bit field, c0 through c7. The following TABLE-12 shows five such code combinations for specifying five different kinds of CLUT update words.

TABLE-12

CT code	Meaning
c0	Write all 3 RGB values into the addressed next-CLUT locations
c1	Write only the BVALUE
c2	Write only the GVALUE
c3	Write only the RVALUE
c4	Write all 3 RGB values into the 33rd background location of the next-CLUT

CADDR (5 bits):

For CLUT update kinds c0 through c3 this field specifies one of 32 entry positions in the next CLUT 235. For CLUT update kind c4, the value is written into the 33rd background color entry of the next CLUT irrespective of this field.

RVALUE (8 bits):

This is the mapped Red value (or Y value or generic A value) to be written into the corresponding R sub-CLUT portion of the next CLUT.

GVALUE (8 bits):

This is the Green conversion value to be written into the corresponding G sub-CLUT portion of the next CLUT.

BVALUE (8 bits):

This is the Blue conversion value to be written into the corresponding B sub-CLUT portion of the next CLUT.

Another type of optional CLUT/video control word is a so-called Video Display Control Word (VDCW). The contents of this control word are loaded into either of VPP registers VDC0 or VDC1 in accordance with the setting of direction bit CN.

Video Display Control Word (VDCW)

3	1	3	3	3	2	2	2	2	2	9
c5	CN	HS	VS	BS	HIE	VIE	FE	DE	MBE	---

When the control type code is CT=c5, the optional control word may be used to set the following control parameters in a designated one of the VDC0 and VDC1 registers.

CN (1 bits):

Control register number. The control signals specified in the rest of this optional word are loaded into one of two control registers (VDC0 or VDC1) identified by this bit. The D-bit in the pixel data then drives a multiplexer (part of which is shown as 435 in Fig. 4A) that selects the output of one of the two control registers (VDC0 or VDC1) for then controlling at least the following functions of the video display path: HS, VS, BS, HIE, VIE, FE, DE, and MBE.

HS (3 bits):

H subposition-bit select. Select source of H bit for Opera style interpolation per TABLE-13.

TABLE-13

HS	Meaning
0	Use D-bit
1	Use Blue Pen 5th msb
2	Set to 0
3	Set to 1
4	Keep previous setting of HS field for this VDC 0/1 register

VS (3 bits):

V subposition-bit select. Select source of V bit for Opera style interpolation per TABLE-14.

TABLE-14

VS	Meaning
0	Use D-bit
1	Use Blue Pen 5th msb
2	Set to 0
3	Set to 1
4	Keep previous setting of VS field for this VDC 0/1 register

BS (3 bits):

Blue bit select. Select source of the blue PEN's 5th msb per below TABLE-15.

TABLE-15

BS	Meaning
0	Use Blue Pen 5th msb
1	Use Green Pen 5th msb
2	Set to 0
3	Set to 1
4	Keep previous setting of BS field for this VDC 0/1 register

For the following 2-bit wide control signals, HIE, VIE, FE, DE, MBE the following encoding of TABLE-16 is used:

TABLE-16

Signal	Meaning
0	Disable
1	Enable
2	Keep previous setting of this field for this VDC 0/1 register

HIE (2 bits): Horizontal interpolation enable.

VIE (2 bits): Vertical interpolation enable.

FE (2 bits): Filter enable.

DE (2 bits): Dither enable.

MBE (2 bits): Matrix bypass enable.

If the control type code CT of an optional CLUT/video update word is equal to a seventh code, c6, the control word functions as an Active Video Display Word (AVDW). This controls display path functions such as horizontal and vertical pixel doubling.

Active Video Display Word (AVDW)

3	11	1	11	1	1	1	1	1	1
c6	HSTART	HSTL	HWIDTH	HWL	HPD	VPD	HPDL	VPDL	---

HSTART (11 bits): Horizontal start of active pixels.

HWIDTH (11 bits): Horizontal width screen line in terms of active pixels.

HPD (1 bit):

Horizontal pixel doubling. When this bit is at logic "1", two active horizontal video pixels will be generated for each frame-buffer pixel. This bit determines how fast pixel data is read out from the FIFO (one or

two frame-buffer pixels per screen pixel). All applications using Opera style interpolation need to fetch two horizontally-adjacent pixels from the frame-buffer and pass them through the CLUT associated with their line (LL or UL) for every produced screen pixel so that, when horizontal interpolation is optionally turned on (by way of HIE), two CLUT-mapped horizontally-adjacent pixels may be simultaneously presented to the interpolator 250 for interpolation.

VPD (1 bit):

Vertical pixel doubling. When this bit is at logic "1", two vertically-adjacent active video lines will be produced for each frame-buffer line. All applications using Opera style interpolation need to be able to produce 2 active vertical video lines for every frame buffer line. Having this bit set implies that vertical interpolation can be turned on (by way of VIE) and two vertically adjacent pixels from the system memory frame-buffer have to be fetched and passed through respective upper and lower CLUT's for each screen pixel that is to be displayed on the VDU screen.

HSTL (1 bit):

HSTART load. If this bit is set, the value in the HSTART field is loaded into the corresponding control register. Else the previous value is kept.

HWL (1 bit):

HWIDTH load. If this bit is set, the value in HWIDTH is loaded into the corresponding

control register. Else the previous value is kept.

HPDL (1 bit):

HPD load. If this bit is set, the value in the HPD field is loaded into the corresponding control register. Else the previous value is kept.

VPDL (1 bit):

VPD load. If this bit is set, the value in the VPD field is loaded into the corresponding control register. Else the previous value is kept.

When the control type CT is equal to a eighth code, c7, the optional word functions as a Video Display List Control Word (VDLCW) having the following structure:

Video Display List Control Word (VDLCW) -Part 1

3	1	1	2	2	1	1	1	1	1
c7	CLUT BP	RGB	CBP SEL	FB FORMAT	FILT TYPE	FTRAN	BKG TRAN	V I L	R D M

Video Display List Control Word (VDLCW) -Part 2

1	1	1	1	1	1	1	11
CLUT BPL	RGB L	CBP SEL L	FB FORMAT L	FILT TYPE L	FTRAN L	BKG TRAN L	---

The fields are as follows:

CLUTBP (1 bit):

CLUT bypass enable. If this bit is set, the soft CLUT's will be bypassed whenever the D-bit in the pixel data is set.

RGB (1 bit):

RGB Mode. Display output format is RGB (=1, disable matrix convert) or YUV (=0).

CBPSEL (1 bit):

CLUT bypass select. When the CLUT's are bypassed, the three least significant bits of each 8-bit color code color may come from four sources. The sources and encoding is specified in the following TABLE-17:

TABLE-17

CBPSEL	Meaning
0	Fill with 0's
1	Fill with 3 msb's from FIFO output
2	Fill with 0's
3	Fill with 3 lsb's from FIFO output

FBFORMAT (2 bits):

Frame buffer format. The video display path can handle three different frame buffer formats. The formats are encoded as shown in below TABLE-18:

TABLE-18

FBFORMAT	Meaning
0	16-bit Opera LR format
1	N/A
2	16-bit M2 format
3	32-bit M2 format

FILTTYPE (1 bit):

Filter type. Selects either the full resolution (=1) or quarter resolution filter (=0).

FTRAN (1 bit):

Force transparency. If this bit is set, the transparency signal sent to the digital encoder will be forced to one for the entire line.

BKGTRAN (1 bit):

Background transparency. If this bit is set, the transparency signal sent to the digital encoder will be asserted whenever a background pixel is detected.

VIL (1 bit):

Vertical interpolation line. If this bit is set, vertical interpolation for the first scanline associated with this VDL will be turned off. The remaining lines associated with this VDL will have vertical interpolation enabled depending on the D-bit and VIE-0/1.

RDM (1 bit):

Random dither matrix index. If this bit is set, a LFSR seed will be used to randomly select which matrix entry to use. If this bit is clear, the row and column position of the pixel within the active display area will be used to index into the dither matrix.

CLUTBPL(1 bit):

CLUTBP load. If this bit is set, load the CLUTBP data into the corresponding control register, else keep previous value.

RGBL (1 bit):

RGB load. If this bit is set, load the RGB data into the corresponding control register, else keep previous value.

CBPSELL (1 bit):

CBPSEL load. If this bit is set, load the CBPSEL data, else keep previous value.

FBFORMATL (1 bit):

FBFORMAT load. If this bit is set, load the FBFORMAT data, else keep previous value.

FILTYPEL (1 bit):

FILTTYPE load. If this bit is set, load the FILTTYPE data, else keep previous value.

FTRANL (1 bit):

FTRAN load. If this bit is set, load the FTRAN data, else keep previous value.

BKGTRANL (1 bit):

BKGTRAN load. If this bit is set, load the BKGTRAN data, else keep previous value.

The above disclosure is to be taken as illustrative of the invention, not as limiting its scope or spirit. Numerous modifications and variations will become apparent to those skilled in the art after studying the above disclosure. For example, the invention is not restricted to RGB formats. Other digital formats such as YCC, or Composite Video Broadcast Standard (CVBS), can also be used. For the sake of simplification, an RGB format was assumed above.

Given the above disclosure of general concepts and specific embodiments, the scope of protection sought is to be defined by the claims appended hereto.

CLAIMS

What is claimed is:

[Note: Bracketed **bold *and italicized* text** is provided in the below claims as an aid for readability and for finding corresponding support in the specification. The bracketed text is not intended to add any limitation whatsoever to the claims and should be deleted in all legal interpretations of the claims and should also be deleted from the final published version of the claims.]

1. A configurable imaging system comprising:

(a) system memory means [*130-131*] for storing data including first data [*140/141*] defining pixels of a frame-buffer and second data [*142/143*] defining display control parameters;

(b) video post-processor means [*150*], coupled to the system memory means and responsive to the display control parameters defined by said second data [*142/143*], the video post-processor means including:

(b.1) first control register means [*VDC0-BS0*] for storing a first setting of control parameters;

(b.2) second control register means [*VDC1-BS1*] for storing a second setting of control parameters;

(b.3) selection means [*435*] responsive to a modulating signal [*D-bit*] that is variable on a pixel-by-pixel basis, said selection means being for selecting one of the first and second control register means [*VDC0/VDC1*] as the control register currently

defining controllable parameters of the video post-processor means [*150*].

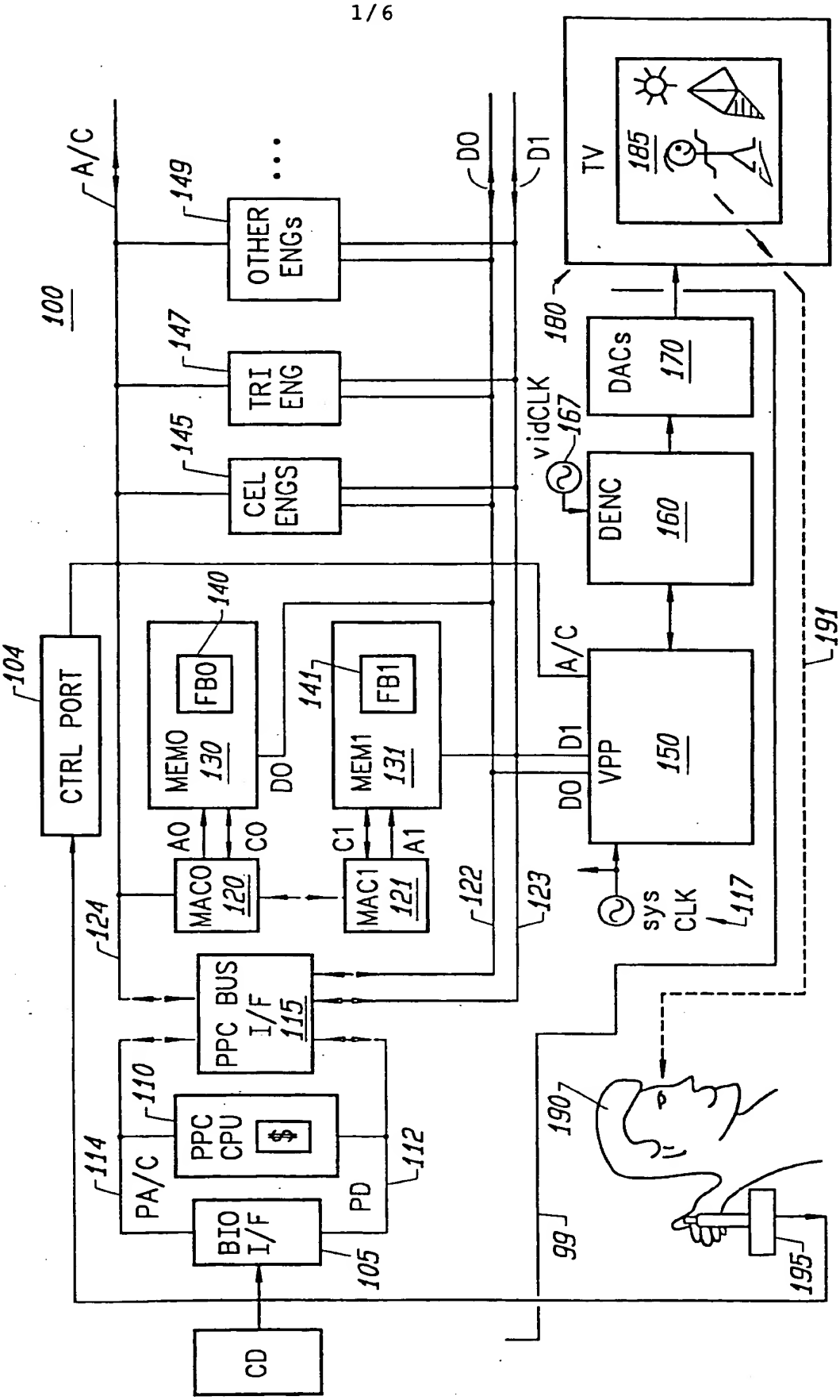


FIG. 1

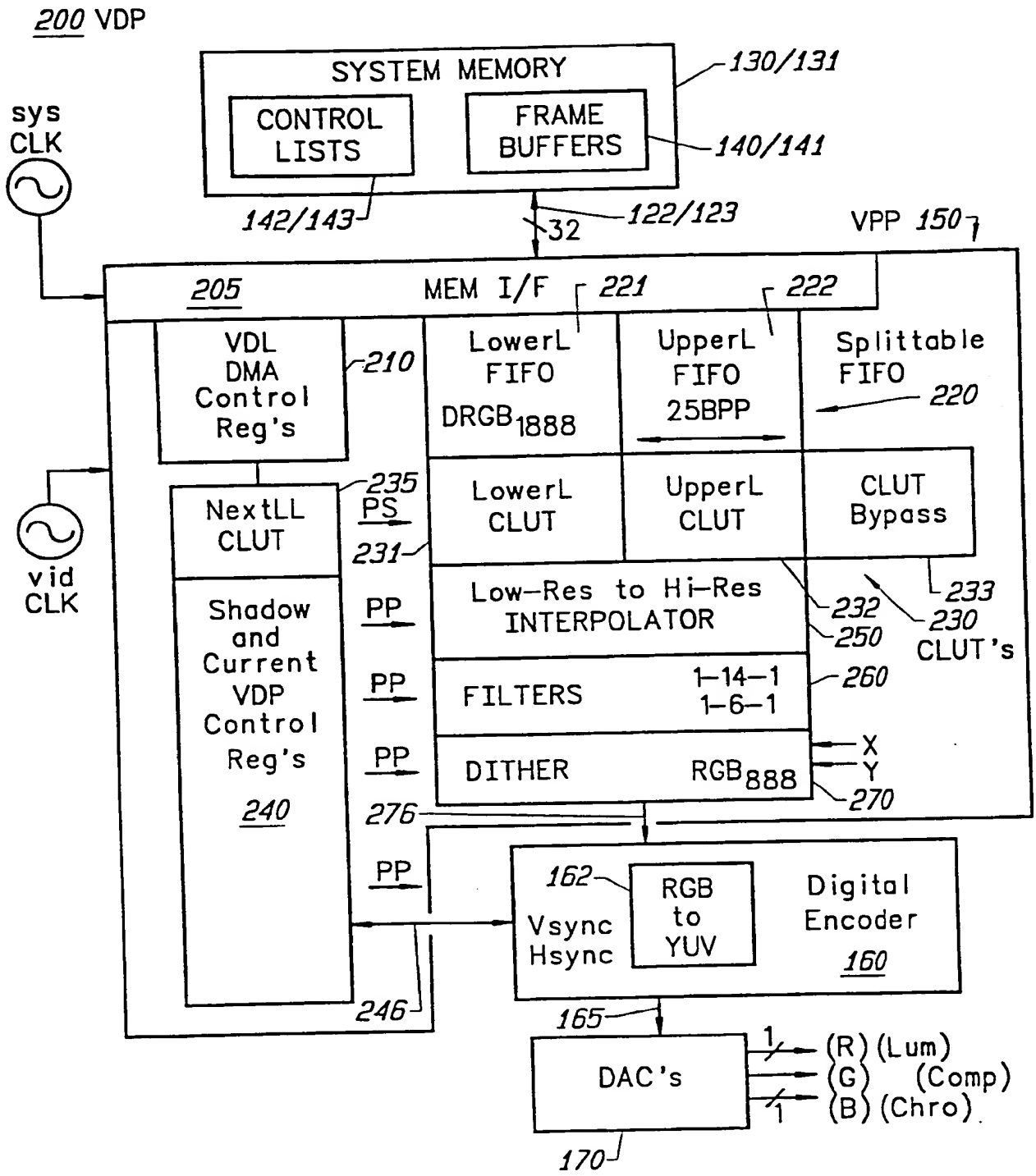


FIG. 2

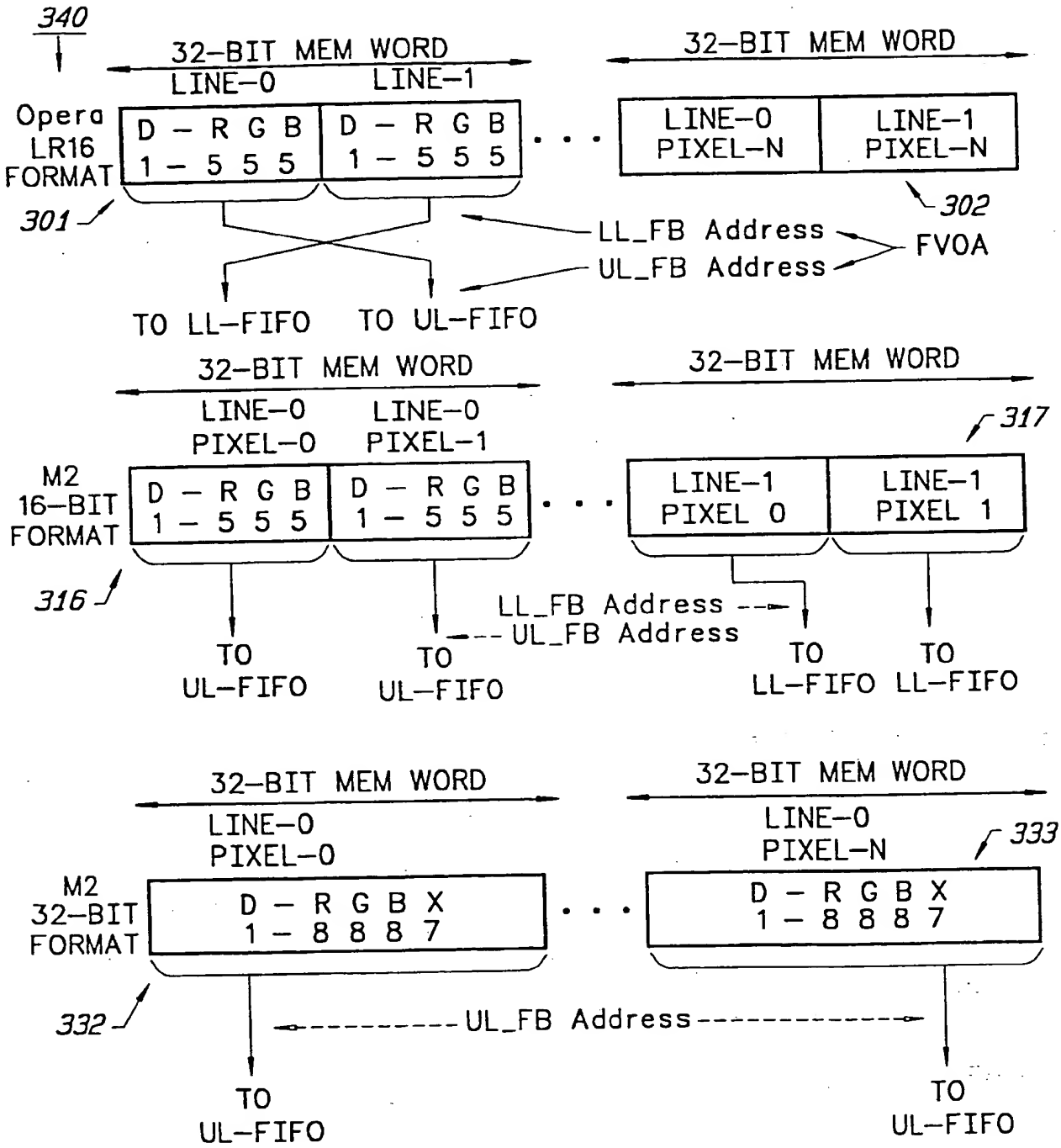


FIG. 3A

SUBSTITUTE SHEET (RULE 26)

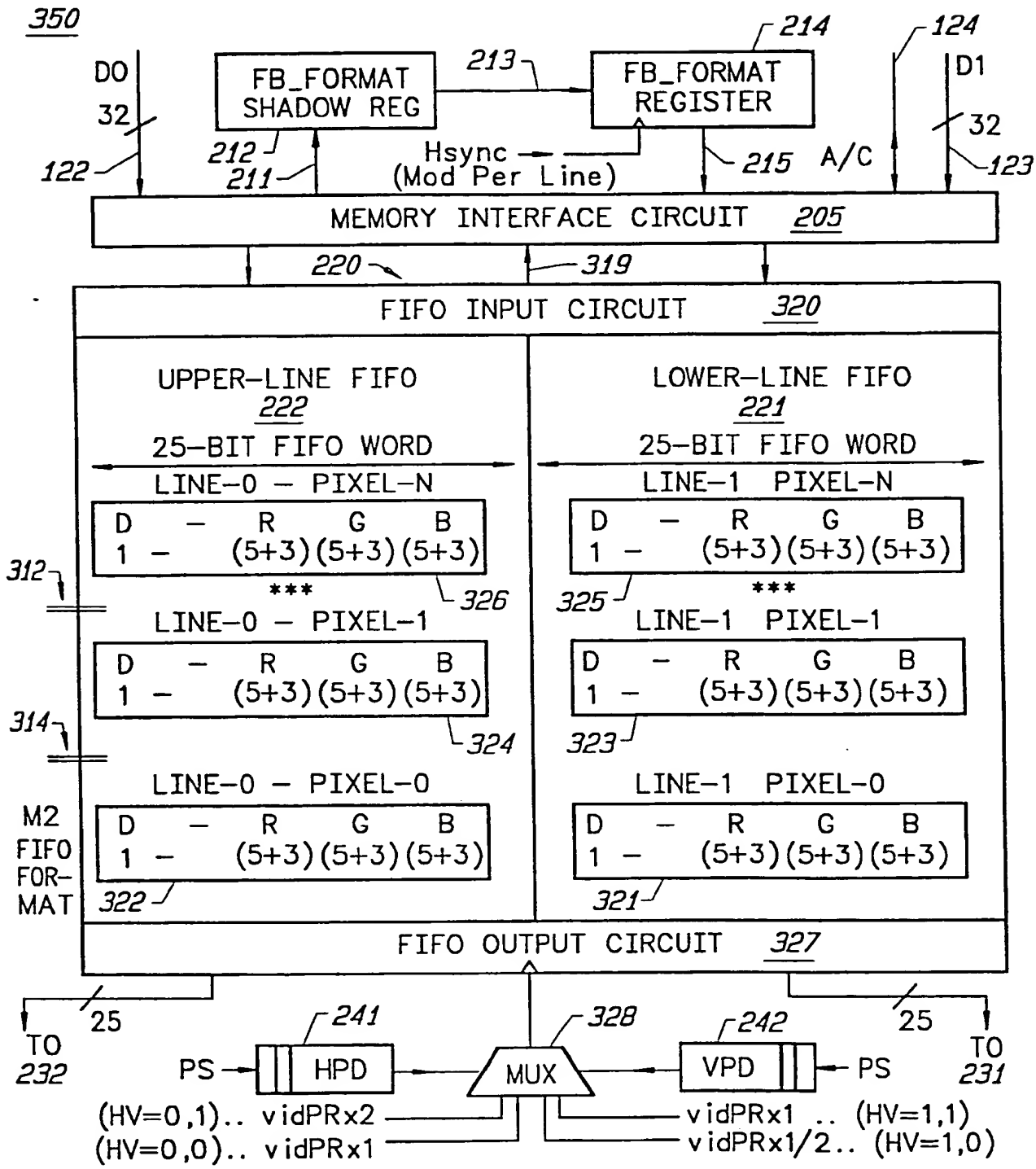


FIG. 3B

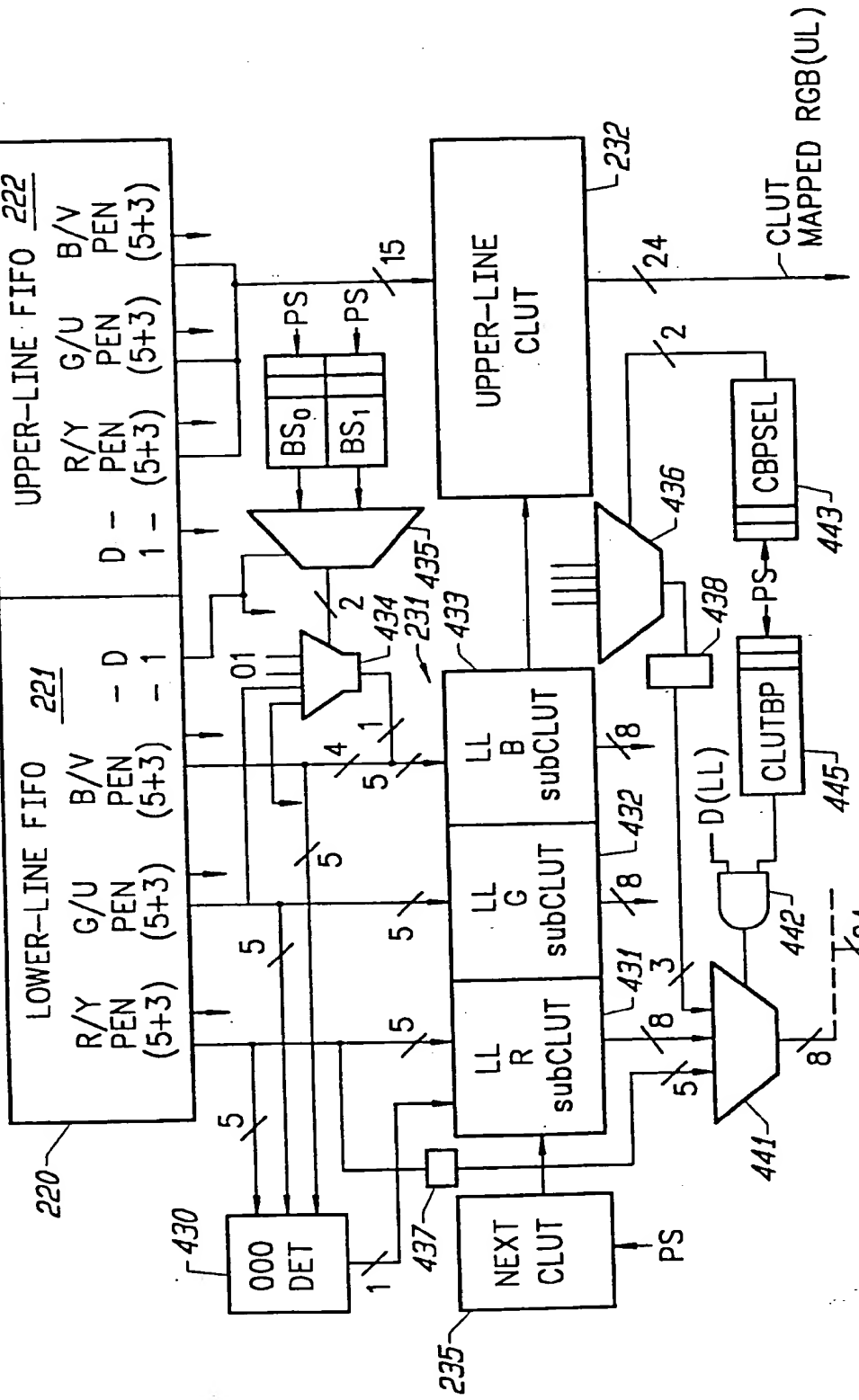


FIG. 4A

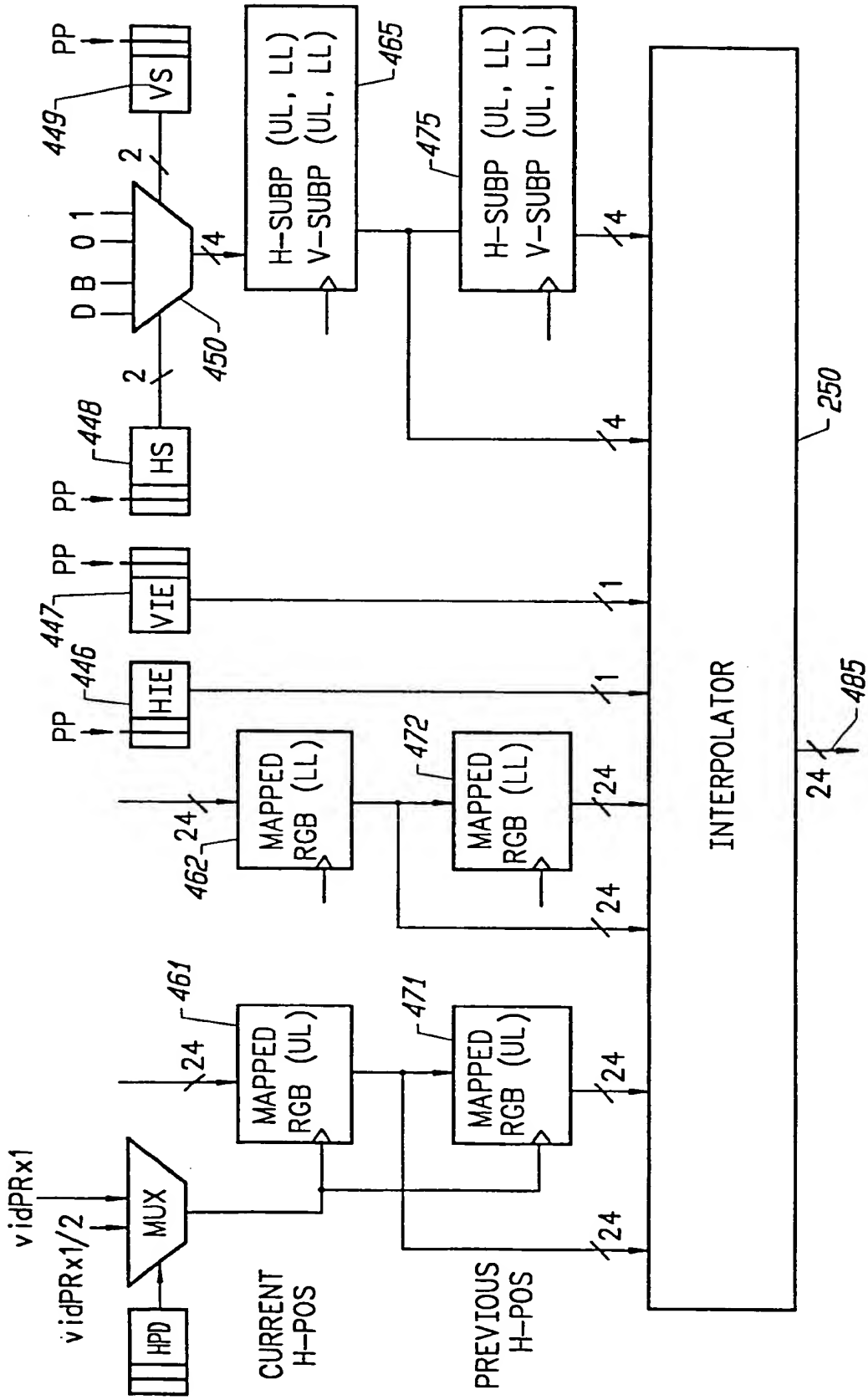


FIG. 4B

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US96/06438

A. CLASSIFICATION OF SUBJECT MATTER
IPC(6) :G09G 5/36
US CL :345/133
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED
Minimum documentation searched (classification system followed by classification symbols)
U.S. : 345/112, 133, 136-139, 150; 348/720, 721; 395/162, 163

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US, A, 4,467,322 (Bell et al) 21 August 1984, note registers 34 and 23, pixel bit line 27, multiplexor 26 and memory 16.	1
Y	US, A, 4,862,150 (Katsura et al) 29 AUGUST 1989, note figure 11.	1

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be part of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 02 JULY 1996	Date of mailing of the international search report 19 JUL 1996
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer JEFFERY A. BRIER Telephone No. (703) 305-4700

Form PCT/ISA/210 (second sheet)(July 1992)*